

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

факультет інформатики та обчислювальної техніки
(повна назва інституту/факультету)

кафедра автоматика та управління в технічних системах
(повна назва кафедри)

«На правах рукопису»
УДК 004.5

«До захисту допущено»

Завідувач кафедри
О. І. Ролік
(підпис) (ініціали, прізвище)

“ ” _____ 2019 р.

Магістерська дисертація

зі спеціальності (спеціалізації) 126 «Інформаційні системи та технології»

на тему: «Інформаційно-довідкова система в кулінарній галузі»

Виконав : студент 6 курсу, групи ІА–382мп
(шифр групи)

Вейдер Дарт Вікторович
(прізвище, ім'я, по батькові) _____ (підпис)

Науковий керівник д.ф.-м.н. професор Дорошенко А.Ю.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) _____ (підпис)

Консультант _____
(назва розділу) _____ (науковий ступінь, вчене звання, прізвище, ініціали) _____ (підпис)

Рецензент _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) _____ (підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

АНОТАЦІЯ

Вейдер Д.В. «Інформаційно-довідкова система в кулінарній галузі», КПІ ім. Ігоря Сікорського, Київ 2019.

Дисертація містить 102 сторінку тексту, 34 рисунків, 46 таблиці, посилання на 56 літературних джерел та 8 додатків.

Метою магістерської дисертації є виявлення актуальних проблем, які виникають при створенні автоматизованих інформаційно-довідкових систем, побудова поліпшеної моделі інформаційно-довідкової системи в кулінарній галузі для підвищення ефективності процесу отримання інформації для користувачів, а також розробка зовнішніх додатків до неї.

Під час роботи над магістерською дисертацією були розглянуті недоліки існуючих інформаційно-довідкових систем.

Завдяки дослідженню даного об'єкту вдалося реалізувати інформаційно-довідкову систему в кулінарній галузі. У пояснювальній записці зазначено етапи створення та налаштування безпеки цієї системи. Для доступу до інформаційно-довідкової системи в кулінарній галузі знадобилося вибрати та реалізувати веб-додаток і додатки для мобільних та планшетних пристроїв.

Для реалізації інформаційно-довідкової системи в кулінарній галузі були проаналізовані і обрані мови програмування, які оптимально підходять для здійснення цього завдання.

Результати даної магістерської дисертації впроваджені на підприємстві ТОВ «Темна сторона сили».

Практичною цінністю результату магістерської дисертації є розробка конкретних аспектів інформаційно-довідкової системи, підвищення ефективності та розширення можливостей використання в сфері роботи кулінарної галузі як за допомогою десктопних пристроїв, так і з мобільних і планшетних пристроїв.

ANNOTATION

Vader DV "Information reference system in the culinary field", KPI named Igor Sikorsky, Kyiv 2019.

The dissertation contains 102 pages of text, 34 figures, 46 tables, links to 56 literary sources and 8 appendices.

The purpose of the master's thesis is to identify actual problems that arise when creating automated information-reference systems, build an improved model of information-reference system in the culinary field to improve the efficiency of the process of obtaining information for users, as well as developing external applications to it.

During the work on the master's thesis, the disadvantages of the existing information and reference systems were considered.

Thanks to the research of this object, we managed to implement a culinary information and reference system. The explanatory note describes the steps in creating and configuring the security of this system. In order to access the culinary information system, it was necessary to select and implement a web application and applications for mobile and tablet devices.

For the implementation of the information and reference system in the culinary field, the programming languages that are best suited for this task have been analyzed and selected.

The results of this master's thesis were implemented at the LTD "Dark Side of Power".

The practical value of the result of the master's thesis is the development of specific aspects of the information-reference system, improving the efficiency and expanding the possibilities of use in the field of work of the culinary industry both with the help of desktop devices, and with mobile and tablet devices.

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра автоматизації та управління в технічних системах

Рівень вищої освіти – другий (магістр) за освітньо-професійною програмою

Спеціальність (спеціалізація) 126 «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О.І. Ролік

«__» _____ 20__ р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Вейдер Дарт Вікторович

1. Тема дисертації «Інформаційно-довідкова система в кулінарній галузі»,

науковий керівник дисертації д.ф-м.н. проф. Дорошенко А.Ю.,

затверджені наказом по університету від «__» _____ 2019 р. №____ -__

2. Термін подання студентом дисертації _____

3. Вихідні дані до розробки: система має відповідати технічному завданню, та має бути зрозумілою кінцевому користувачеві.

4. Зміст пояснювальної записки

Перелік термінів, вступ, огляд існуючих рішень, використані технології, опис програмного продукту, опис інтерфейсу системи, створення стартапу, висновки, перелік використаних джерел.

5. Перелік графічного матеріалу

Діаграма структури та зв'язків таблиць бази даних, діаграма структури та зв'язків таблиць бази даних, структурна схема web – системи, UseCase – діаграма, діаграма послідовності дій користувача, діаграма розгортання, Функціональна схема, UML-діаграма системи обліку робочого часу.

6.Консультанти розділів дисертації

Розділ	Прізвище, ініціали, та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів дисертації	Примітка
1.	Огляд існуючих рішень	05.09.19 – 06.09.19	
2.	Структура системи	07.09.19 – 08.09.19	
3.	Конструювання бази даних	09.09.19 – 16.09.19	
4.	Вибір мов програмування	17.09.19 – 19.09.19	
5.	Створення системи	20.09.19 – 30.11.19	
6.	Оптимізація системи для планшетних і мобільних пристроїв	01.12.19 – 03.12.19	
7.	Оформлення пояснювальної записки	04.12.19 – 15.12.19	

Студент

Д. В. Вейдер

Керівник дисертації

А. Ю. Дорошенко

ЗМІСТ

ПЕРЕЛІК ТЕРМІНІВ ТА СКОРОЧЕНЬ	6
ВСТУП	11
1 ОГЛЯД ІСНУЮЧИХ СИСТЕМ.	13
1.1 Характеристики структур систем.	13
1.2 Огляд існуючих систем	15
1.2.1 Система russianfood.com.	15
1.2.2 Система povarenok.ru.	17
1.2.3 Система gotovim-doma.ru.	18
1.3 Висновок до розділу	20
2 СТРУКТУРА СИСТЕМИ	22
2.1 Історія створення та розвитку інформаційної довідкової системи.	22
2.2 Вимоги до системи.	24
2.3 Склад зовнішньої і внутрішньої структури системи .	26
2.4 Структура системи - змішана структура	27
2.5 Вимоги до розділів системи.	33
2.6 Висновок до розділу	36
3 КОНСТРУЮВАННЯ БАЗИ ДАНИХ.	37
3.1 Види систем управління базами даних.	37
3.2 MySQL - вільна реляційна система управління базами даных	39
3.3 Створення бази даних в MySQL	40
3.4 Структура таблиць і їх внутрішній зміст	42
3.5 Висновки до розділу	55
4 ВИБІР МОВ ПРОГРАМУВАННЯ.	56
4.1 Вимоги для проектування веб системи.	56
4.2 Аналіз мов і методів для створення системи	56

4.2.1 Дослідження мов програмування веб-системи. . . .	56
4.2.2 CSS3 - каскадна таблиця стилів.	58
4.2.3 JavaScript - мова програмування.	58
4.2.4 Мова AJAX технологія звернення до сервера без перезавантаження сторінки	59
4.2.5 Текстовий формат обміну даними JSON.	59
4.2.6 jQuery - бібліотека <i>JavaScript</i>	60
4.3 Опробування системи.	60
4.3.1 Опробування функцій системи.	60
4.3.2 Перевірка верстки на різних пристроях.	61
4.3.3 Випробування безпеки.	62
4.3.4 Тестування ефективності системи.	63
4.4 Висновок до розділу	64
5 СТВОРЕННЯ СИСТЕМИ	65
5.1 Висновок до розділу	68
6 ІНСТРУКЦІЯ КОРИСТУВАЧА	69
6.1 Висновок до розділу	71
7 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ.	72
7.1 Історія створення та розвитку стартапів	73
7.2 Загальна інформація про стартап	74
7.3 Інформаційна карта проекту	74
7.4 Команда стартап-проекту	76
7.5 Маркетинговий аналіз стартап-проекту	77
7.5.1 Маркетингова концепція товару	77
7.5.2 Опис ідеї проекту	77
7.5.3 Технологічний аудит ідеї проекту	80
7.5.4 Маркетинговий план системи	81
7.5.5 Впізнаваність системи	82

7.5.6 Аналіз ринкових можливостей запуску стартап-проекту	85
7.5.7 Розроблення ринкової стратегії проекту	92
7.5.8 Розроблення маркетингової програми стартап-проекту	95
7.6 Висновок до розділу	96
ВИСНОВКИ	97
Перелік використаних джерел.	98

ПЕРЕЛІК ТЕРМІНІВ ТА СКОРОЧЕНЬ

Веб-система, або веб-ресурс — це сторінка або кілька сторінок, розміщених в мережі Інтернет. Такі сторінки можуть включати текстову, графічну інформацію або мультимедіа-компоненти. Сторінки веб-ресурсу можуть бути як статичними (plain-HTML), так і динамічними.

Дизайн системи — це сукупність графічних елементів, шрифтів і кольорів, реалізованих на сайті.

Header (хедер, Хідер, шапка) — це блок у верхній частині сторінки сайту, в якому, як правило, розміщується логотип і слоган сайту, коротка контактна інформація, основне горизонтальне меню та інші елементи, які вважаються найбільш важливими в залежності від специфіки ресурсу.

Footer (футер, підвал) — футер, він же підвал сайту - це блок в нижній частині сторінки, куди виносять корисну, але не першорядну інформацію. Як приклади можна навести дані про копірайт і дублювання основних пунктів навігації.

Sitebar (сайтбар) — це колонка сайту, що включає в себе основні навігаційні елементи і важливі для відвідувача блоки. Прикладом сайтбара може служити ліва частина сайту, що включає меню сайту, категорії, рекламний блок і інформер зі статистикою сайту.

Контент (калька з англ. Content - зміст, вміст; іноді використовують - інформаційне наповнення, наповнення) — збірний термін, що характеризує будь-яку інформацію, яка міститься на сторінках веб-ресурсу. У буквальному сенсі контентом можна назвати тексти, аудіо та відеофайли, графічні зображення, анімацію, картинки та іншу інформацію, розміщену на інтернет-ресурсі (все, що користувач може прочитати, побачити, почути).

Меню сайту — це згрупований набір посилань з назвами розділів, що полегшує перехід на інші сторінки. Вони називаються пунктами меню і можуть позначатися текстом або графічними значками — іконками.

Б/Ж/В — білків, жирів, вуглеводів

Ккал — калорій

Браузер (від англ. Web Browser) — прикладне програмне забезпечення для перегляду веб-сторінок, змісту веб-документів, комп'ютерних файлів та їх каталогів; управління веб-додатками; а також для вирішення інших завдань.

phpMyAdmin (надалі РМА) - СУБД — додаток з відкритим кодом, написаний на мові PHP і представляє собою веб-інтерфейс для адміністрування СУБД MySQL. phpMyAdmin дозволяє через браузер і не тільки здійснювати адміністрування сервера MySQL, запускати команди SQL і переглядати вміст таблиць і баз даних. Додаток користується великою популярністю у веб-розробників, так як дозволяє управляти СУБД MySQL без безпосереднього введення SQL команд, надаючи дружній інтерфейс.

HTTP (абр. від англ. Hyper Text Transfer Protocol) — являє собою протокол передачі даних для інформаційних систем. HTTP є основою передачі даних для інтернету.

HTTPS (абр. від англ. HyperText Transfer Protocol Secure) — розширення протоколу HTTP для підтримки шифрування з метою підвищення безпеки. Дані в протоколі HTTPS передаються поверх криптографічних протоколів SSL або TLS.

SSL (абр. від англ. Secure Sockets Layer) — рівень захищених сокетів.

SQL (абр. від англ. Structured Query Language) — спеціальна мова, що використовується для роботи в реляційних БД.

MySQL — вільна система керування реляційними базами даних.

PHP (абр. від англ. PHP: Hypertext Preprocessor) — «PHP: препроцесор гіпертексту». Спочатку Personal Home Page Tools - «Інструменти для створення персональних веб-сторінок»). Скриптова мова загального призначення, інтенсивно застосовується для розробки веб-додатків.

JavaScript — мультіпарадігменна мова програмування. Підтримує об'єктно-орієнтований, імперативний і функціональний стилі. Є реалізацією мови ECMAScript (стандарт ECMA-262).

JSON (абр. від англ. JavaScript Object Notation) — текстовий формат обміну даними, заснований на JavaScript.

jQuery — бібліотека фокусується на взаємодії JavaScript і HTML, також надає зручний API для роботи з AJAX.

API (абр. від англ. Application Programming Interface) — програмний інтерфейс програми, інтерфейс прикладного програмування. набір готових класів, процедур, функцій, структур та констант, що надаються додатком (бібліотекою, сервісом) або операційною системою для використання у зовнішніх програмних продуктах. Використовується програмістами при написанні всіляких додатків.

AJAX (абр. від англ. Asynchronous Javascript and XML) — асинхронний JavaScript та XML, підхід до побудови призначених для користувача інтерфейсів веб-додатків, що полягає в «фоновому» обміні даними браузера з веб-сервером. В результаті, при оновленні даних веб-сторінка не перезавантажується повністю, та веб-додатки стають швидше та зручніше.

CSS (абр. від англ. Cascading Style Sheets) — каскадні таблиці стилів, формальна мова опису зовнішнього вигляду документа, написаного з використанням мови розмітки.

XSS (англ. Cross-Site Scripting - «міжсайтовий скриптинг») — тип атаки на веб-системи, що полягає у впровадженні в видавану веб-системою сторінку шкідливого коду (який буде виконаний на комп'ютері користувача при відкритті їм цієї сторінки) і взаємодії цього коду з веб-сервером зловмисника. Є різновидом атаки «впровадження коду».

SQL ін'єкція (англ. SQL injection) — один з поширених способів злому сайтів і програм, які працюють з базами даних, заснований на впровадженні в запит довільного SQL-коду.

JAVA — строго типізований об'єктно-орієнтована мова програмування, який був розроблений компанією Sun Microsystems (потім був придбаний компанією Oracle).

SWIFT — відкрита мультипарадигмальності компільована мова програмування загального призначення. Був створений компанією Apple для розробників iOS і macOS. Мова працює з фреймворками Cocoa і Cocoa Touch і сумісна з основною кодовою базою Apple, яка написана на Objective- C.

Фронтенд (англ. Front-end) — клієнтська сторона призначеного для користувача інтерфейсу до програмно-апаратної частини сервісу.

Бекенд (англ. Back-end) — програмно-апаратна частина сервісу.

Фронт і бекенд — це варіант архітектури програмного забезпечення.

Терміни виникли в програмній інженерії внаслідок розвитку принципу поділу відповідальності між зовнішнім поданням і внутрішньої реалізацією.

Back-end для надання своєї функції реалізує API, яке використовує front-end.

Медійна реклама - направлена на привернення уваги аудиторії до продукту і підштовхує продажу, як в онлайн, так і офлайн. Навіть, якщо споживач не відреагував на неї відразу, він може згадати про неї пізніше. До медійної реклами відносяться листівки, банери на сайті, рекламні ролики на телебаченні, біг-борди.

Цифровий маркетинг (англ. Digital marketing, Діджитал-маркетинг) - термін, який вживається для визначення таргетівного і інтерактивного маркетингу товарів і послуг. Для залучення клієнтів і утримання їх як споживачів цифровий маркетинг застосовує цифрові технології.

Конверсія в інтернет-маркетингу - це відношення числа відвідувачів сайту, які виконали на ньому якісь цільові дії (явні або неявні розпорядження рекламодавців, продавців, творців контенту - придбання, реєстрацію, підписку, відвідування якийсь сторінки сайту, перехід по рекламному посиланню), до загальної кількості відвідувачів сайту, показане у відсотках.

Таргетінг (англ. target – ціль) – рекламна конструкція, яка робить можливим виділення зі всієї наявної аудиторії тільки її частину, яка задовольняє заданим критеріям (цільову аудиторію), і продемонструвати рекламу саме їй.

КМС (або контекстно-медійна мережа) – Google являє собою величезну мережу партнерських майданчиків, де вебмастери розміщують блоки контекстної реклами.

Зовнішній контент-маркетинг - мікс піару і контент-маркетингу. Один з найпотужніших інструментів для підвищення впізнавання бренду з єдиною умовою - він повинен бути розташований там, де його зможе побачити цільова аудиторія.

Охоплення (англ. Reach) - кількість людей або домогосподарств, в процентному співвідношенні або абсолютних цифрах, які мали контакт з подією хоча б один раз.

SEO (Search Engine Optimization) – комплекс дій, що допомагають потенційним клієнтам відразу ж знайти ваш сайт в пошукових системах Google, Яндекс (і інших). Мета SEO - організувати довіру роботів до ресурсу.

Внутрішній контент-маркетинг - формування бренду, визначаючись на образ представника цільової аудиторії, звичайний для нього спосіб спілкування і отримання інформації.

Лідер думки - це людина, яка відрізняється високим соціальним статусом і кращої поінформованістю в очах своїх послідовників і тому, має вплив на розуміння цих людей змісту і сенсу повідомлень масової комунікації.

ВСТУП

Сьогодні комп'ютерні технології впроваджуються практично в усі сфери життєдіяльності людини. Виробництво, документообіг немислимі без використання обчислювальної техніки. Потрібно відзначити ряд переваг комп'ютеризації робочих процесів будь-якої організації:

- 1) можливість мати під рукою великий обсяг потрібної інформації;
- 2) мати можливість швидкого доступу до необхідної інформації;
- 3) можливість генерувати всілякі звіти;
- 4) можливість завжди мати актуальну базу даних.

Усе це накладає на процес автоматизації ряд системних вимог на початкових етапах. Необхідно чітко і повно змальовувати завдання, що впливають з проведеного аналізу роботи в даній сфері виробництва.

Актуальність магістерської дисертації характеризується її затребуваністю при стрімкому розвитку комп'ютерних технологій інформаційних систем, а також програмних засобів. Вимога і важливість вирішення таких завдань на сучасному етапі і породило провести дослідження існуючих такого роду інформаційно довідкових баз для зниження трудомісткості процесів їх використання. Сьогодні інформаційні технології роблять роботу в сфері кулінарії якісніше і доступніше для великої кількості людей, що цікавляться.

Довідкові системи дозволяють впоратися з будь-якими завданнями для задоволення запитів споживачів в отриманні необхідної інформації.

Пошук необхідної інформації в інформаційно довідкових системах відбувається значно швидше, ніж в паперовому аналогу. Сьогодні налічується величезна кількість інформаційно довідкових систем з кулінарної тематики.

Об'єктом дослідження є розробка автоматизованих інформаційних систем в кулінарної галузі.

Предметом дослідження є розробка автоматизованої інформаційної системи рецептів кулінарних блюд і зовнішнього додатка до неї.

Метою магістерської дисертації є розробка поліпшеної моделі інформаційно - довідкової системи в кулінарній галузі для підвищення ефективності процесу отримання інформації для користувачів, а також розробка зовнішнього додатка до неї на основі виявлених актуальних проблем, що виникають при створенні автоматизованих інформаційних систем.

Методи дослідження:

- 1) знайти і описати вимоги до системи;
- 2) реалізувати проектування бази даних;
- 3) реалізувати базу даних;
- 4) здійснити зовнішній додаток до бази даних.

Для досягнення зазначеної мети необхідно розв'язати наступні завдання:

- інформаційно довідкова система в кулінарній галузі повинна мати зручне поповнення бази;
- необхідно розглянути і проаналізувати існуючі технології, протоколів і стандартів в цій області;
- розробити досконаліший алгоритм знаходження потрібних рецептів з урахуванням заданих інгредієнтів;
- розробити базу даних і програмний комплекс для роботи з довідковою системою;
- перевірити ефективність створених алгоритмів і розробленої інформаційно-довідкової системи

Практичною цінністю результату магістерської дисертації є застосування інформаційно довідкової системи в кулінарній галузі як за допомогою десктопних пристроїв, так і з мобільних і планшетних пристроїв.

1 ОГЛЯД ІСНУЮЧИХ СИСТЕМ

Зараз в інтернеті можна знайти величезну кількість кулінарних порталів, в яких є безліч різних кулінарних рецептів різноманітних блюд і люди користуються такими послугами. Це зручно, не займає багато часу і дає великий вибір рецептів. Але що дійсно хочуть вони побачити в цих рецептах? Які параметри їх цікавлять? Який сервіс має бути на цьому сайті?

1.1 Характеристики структур систем

Нижче розглянуто певні структури сайтів.

Лінійна структура. Це найпростіша структура сайту, де Web- сторінки йдуть одна за одною, і користувач повинен переглядати їх як слайд-шоу. Тут немає розділення контенту на рівні. Усі сторінки тут рівноправні, і їх бачить будь-який відвідувач сайту.

Лінійна з відгалуженнями. Та ж лінійна структура з відгалуженнями, що відбуваються з різних внутрішніх сторінок web – системи. Тобто відвідувач також послідовно переходить з однієї сторінки на іншу, але має можливість при необхідності переходити на відгалуження, а потім повертатися назад.

Блокова структура - це структура, в якій всі сторінки посилаються на декілька інших, рівнозначних між собою.

Деревоподібна структура. Сенс застосування такої структури в тому, що у клієнта є можливість переходу у будь-який розділ, підрозділ і на конкретну сторінку як з головної сторінки сайту, так і будь-який інший.

Змішана структура. Для цієї структури характерно те, що в ній можуть бути присутніми елементи двох або більше перерахованих вище структур.

Ця структура складніше усіх описаних вище. Тут усі сторінки також розміщуються в різних гілках. Проте користувач має можливість переміщатися по них як вертикально (вгору-вниз), так і горизонтально (тобто між гілками на різних рівнях).

Переваги та недоліки структур систем

Лінійна структура. Перевагою є те, що усі сторінки тут йдуть послідовно одна за одною, як сторінки в звичайної книги, і тому має обмежені можливості.

До недоліків відноситься, те що тут є всього три посилання: на попередню, наступну і на головну сторінку. Тут не має розділів, і користувач повинен рухатися по сайту, так, ніби читає книжку, не перестрибуючи по сторінках в будь-якій послідовності. Але цим утруднюється пошук необхідної інформації.

Лінійна з відгалуженнями. Перевагою є те, що ця структура, є лінією з нескінченної кількості невеликих відгалужень, де користувач може детальніше подивитися іншу інформацію, і продовжити працювати з інформацією основної лінії. Плюсом в цій структурі також є те, що до неї можна просто перейти із звичайної лінійної структури, з незначними модифікаціями.

Істотним недоліком цієї структури є те, що то при розростанні системи виникає необхідність поліпшення навігації.

Блокова структура. Перевагою цієї структури є, то всі сторінки посилаються на кілька інших, рівнозначних між собою.

Великим недоліком цієї структури є неприпустимість розвивати і масштабувати систему.

Деревоподібна структура. Головне достоїнство цієї структури – її універсальність. Вона може підійти для будь-якого виду сайту. Тут прекрасна навігація, тобто клієнт має вибір і можливість переходити з будь-якої сторінки сайту у будь-який розділ, підрозділ або конкретну сторінку.

Недоліком є те, що використовуючи деревовидну структуру складно дотримуватися балансу між "глибиною і шириною". І якщо "дерево" сайту буде рости тільки углиб, то користувачам, доведеться завантажити і розглянути надто багато сторінок, щоб дійти до якоїсь інформації, а це створює незручності для користувачів.

При створенні дуже широкої деревовидної структури відвідувачі повинні будуть кожного разу витратити багато часу для вибору потрібної їм гілки, що

теж не доцільно. І це теж погано. Тому, при використанні деревовидної структури сайту обов'язково потрібно стежити за її розростанням і дотримуватися золотой середини.

Змішана структура. Перевагою змішаних структур є те, що вони складаються з декількох компонентів описаних вище структур, з'єднаних в одну. Отже, така структура складніше описаних вище. Всі її сторінки розміщуються на різних відгалуженнях. Пересування між відгалуженнями на глибинному рівні здійснюються за допомогою посилань на рубрики інших розділів.

Недоліки усуваються за допомогою поєднанням інших вищеописаних структур.

1.2 Огляд існуючих систем

Для того, щоб провести аналіз існуючих інформаційно-довідкових систем в кулінарній галузі візьмемо 3 найпопулярніших системи в цій галузі russianfood.com, povarenok.ru і gotovim-doma.ru

1.2.1 Система russianfood.com

Система russianfood.com має зрозумілий інтерфейс. На головній сторінці видно розбивка по розділах типів страв. Система має досить велику базу кулінарних рецептів, в тому числі і вибір кухонь народів світу, багато рецептів представлені з фотографіями, відео і покроковим описом приготування.

Тут є зручна система пошуку.

Система має досить велику кількість розділів кухарський книги. Частина рецептів забезпечена фотографіями, на деякі рецепти надані відео, є рецепти з прикріпленим покроковому описом дій. Також є перелік пропонованих інгредієнтів із зазначенням їх кількості.

Скріншот головної сторінки системи russianfood.com зображений на рисунку 1.1. [1]

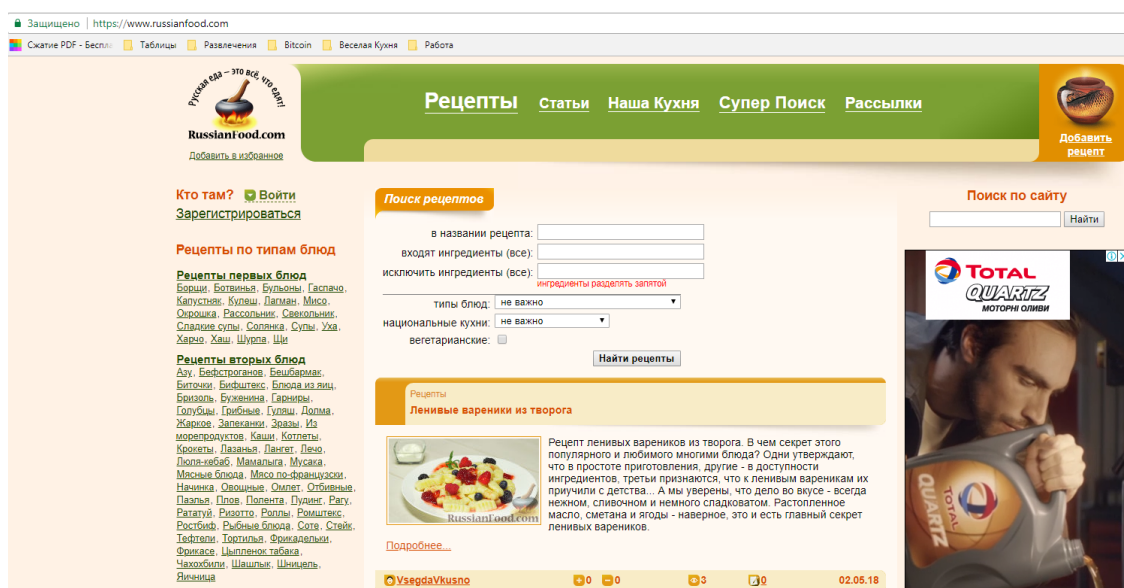


Рисунок 1.1 - Головна сторінка russianfood.com

Система має деревоподібну структуру.

Структура системи russianfood.com зображена на рисунку 1.2.

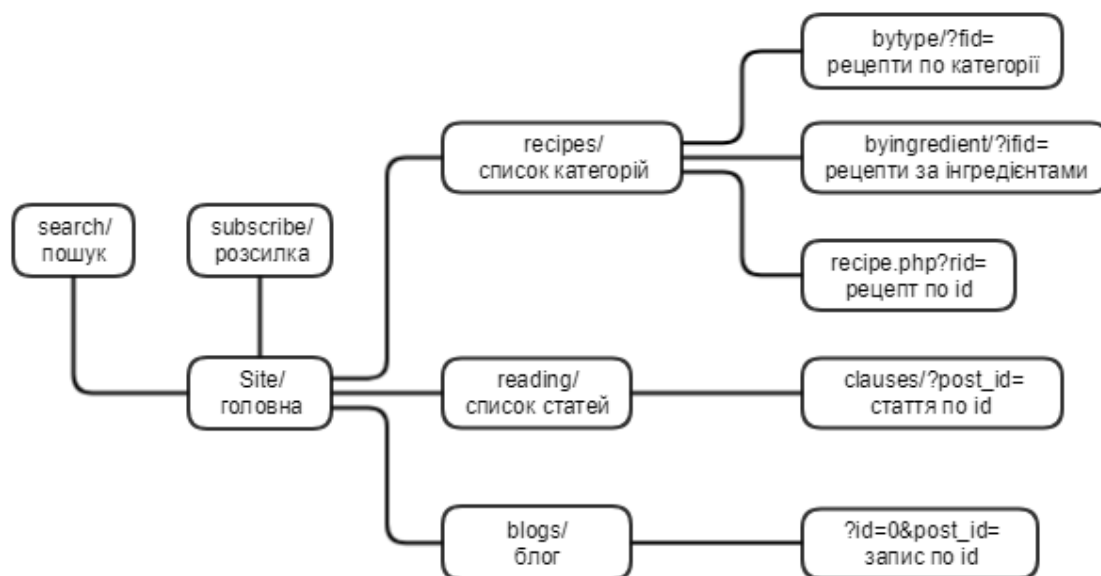


Рисунок 1.2 – Структура системи russianfood.com

1.2.2 Система povarenok.ru

В системі показаний каталог рецептів. Кожен рецепт має фото і опис. Виконано хороший перегляд рецептів (є опис, вказані час приготування і кількість порцій, необхідні інгредієнти). На частину рецептів дані відео рецепти. Виконано сортування за рейтингом, датою і датою додавання в кулінарну книгу і навіть результатам пошуку. Можна побачити в системі корисні поради користувачів і розсилки, на які пропонують підписатися.

Є допомога вертикального і горизонтального режиму. Частина рецептів представлена з відео

Скріншот головної сторінки системи povarenok.ru. зображений на рисунку 1.3. [2]

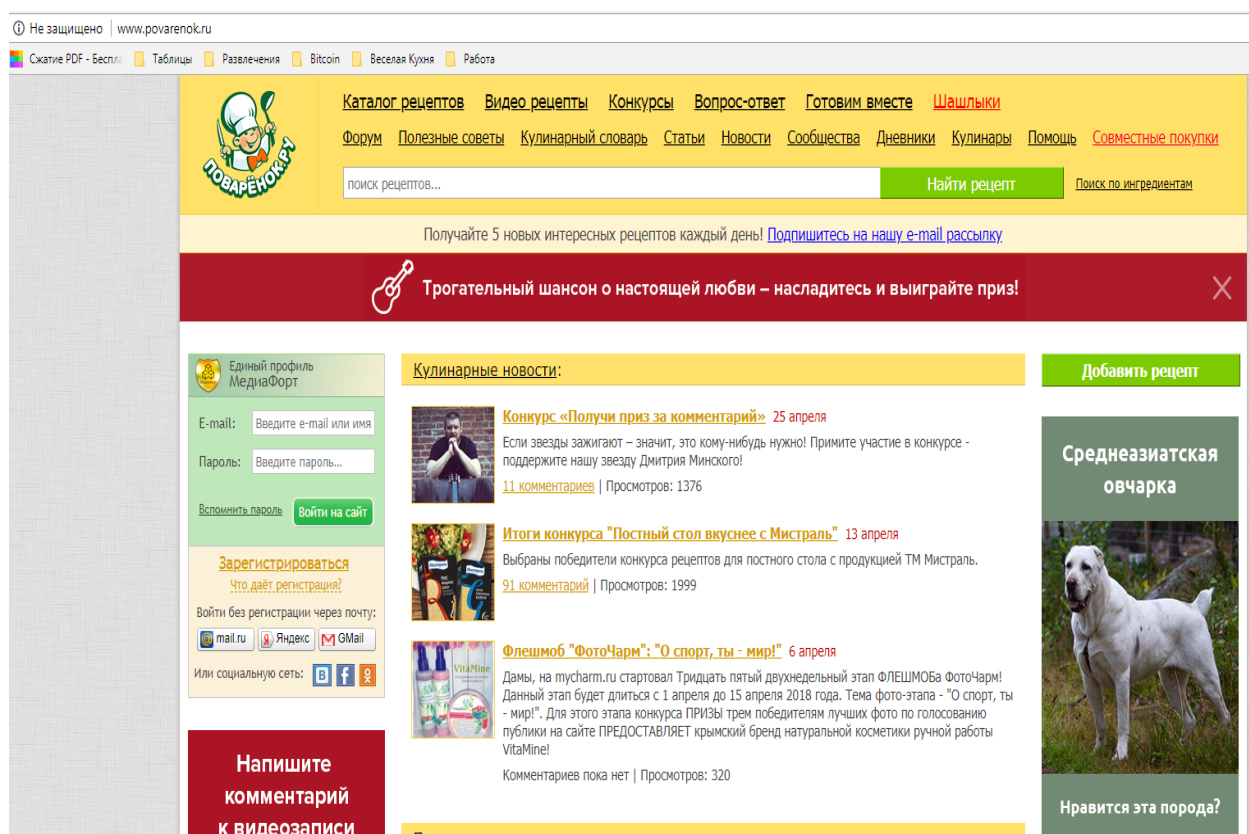


Рисунок 1.3- Головна сторінка povarenok.ru

Система має деревоподібну структуру.

Структура системи rovarenok.ru. зображена на рисунку 1.4.

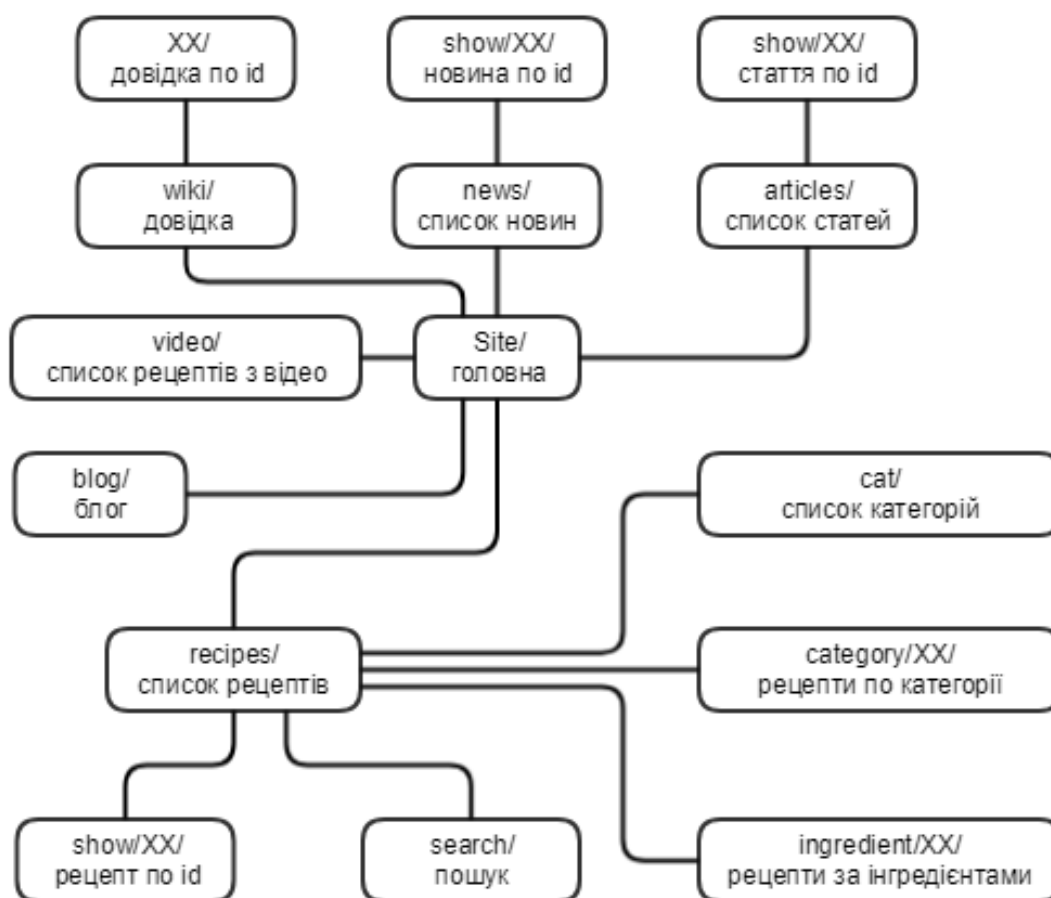


Рисунок 1.4 – Структура системи rovarenok.ru

1.2.3 Система gotovim-doma.ru

Хоча ця система була створена в 2003 році, перший користувач на форумі зареєструвався в 2006 році. В системі зроблена розбивка по розділах.

Головне меню і навігаційне меню показується на всіх сторінках. Це дозволяє легко і швидко здійснювати перехід між сторінками. Надана інформація - кулінарні рецепти - відповідає звичайним запитам користувача. На деякі рецепти дано покроковий опис дій.

Частина рецептів мають відео.

Вказані пропоновані інгредієнт і їх кількість. Але програма не реалізує функцію підбору поєднаних інгредієнтів.

Немає можливості зробити підбір персонального раціону на тиждень.

Немає підрахунку калорій і Б/Ж/В страв.

Система має свій форум, де зареєстровані користувачі при бажанні можуть брати участь. Реалізовано пошук страв за складовими.

Скріншот головної сторінки системи gotovim-doma.ru. зображений на рисунку 1.5. [3]

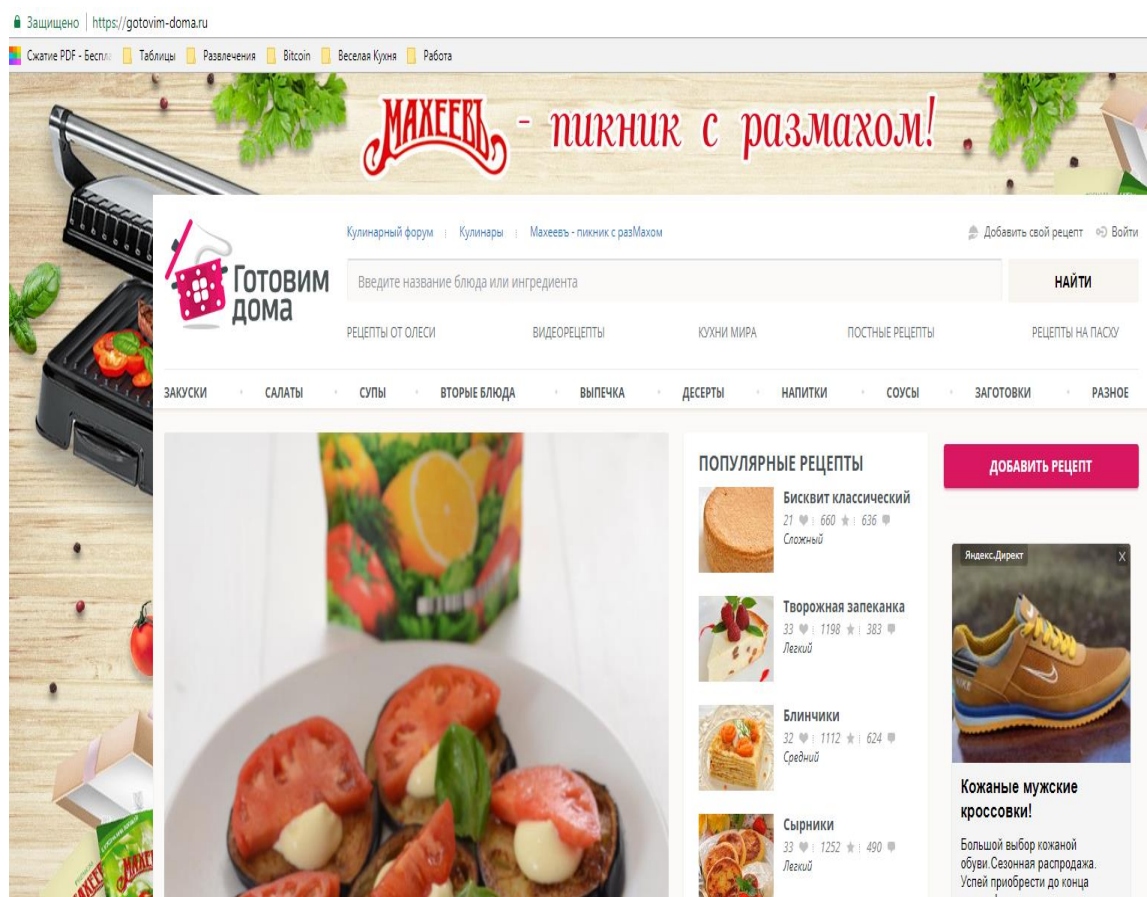


Рисунок 1.5 - Головна сторінка gotovim-doma.ru

Система має лінійну з відгалуженнями структуру.

Структура системи gotovim-doma.ru зображена на рисунку 1.6.

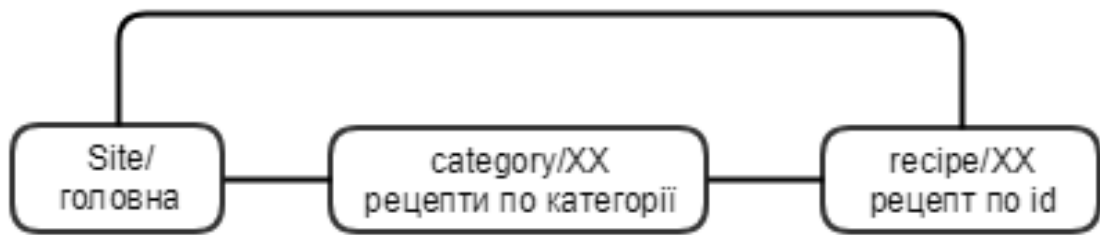


Рисунок 1.6 – Структура системи gotovim-doma.ru

1.3 Висновок до розділу

Були розглянуті 3 системи з кулінарії галузі та проведено порівняльний аналіз за такими параметрами:

- зручність інтерфейсу;
- кількість користувачів в день;
- наявність інформації щодо кількості Б / Ж / У і ккал в страві;
- наявність інформації про інгредієнти;
- структура системи;
- наявність SSL сертифікату;
- підтримка мобільних і планшетних пристроїв;
- присутність реклами.

Внаслідок був зроблений наступний висновок: кращою системою виявилась russianfood.com, проте є недоліки і у нього.

У таблиці 1.1 відображена інформація порівняльного аналізу 3 систем.

Таблиця 1.1 – Порівняльна таблиця аналізу існуючих систем

Система	russianfood.com	povarenok.ru	gotovim-doma.ru
Кількість користувачів у день	400 тис.	250 тис.	50 тис.
Структура системи	Деревоподібна структура	Деревоподібна структура	Лінійна з відгалуженнями

Продовження таблиці 1.1

Система	russianfood.com	povarenok.ru	gotovim-doma.ru
Наявність SSL сертифікату	Є	Немає	Є
Наявність інформації про кількість Б/Ж/В та ккал у страві	Відсутня	Є	Відсутня
Зручність інтерфейсу	Зручний	Зручний	Незручний
Присутність реклами	Забгато	Забгато	Забгато
Наявність інформації про інгредієнти	Відсутня	Відсутня	Відсутня
Наявність підтримки мобільних і планшетних пристроїв	Підтримує	Підтримує	Не підтримує

З таблиці видно, що досліджувані системи мають майже однакові недоліки, хоча вони мають різні структури.

2 СТРУКТУРА СИСТЕМИ

2.1 Історія створення та розвитку інформаційної довідкової системи.

Інформаційно-довідкова система - система реєстрації, накопичення, переробки та зберігання інформації, яка створена для забезпечення користувачів даними довідкового характеру. Класичною інформаційно-довідковою системою є книга. Вона залишається важливою і сьогодні, але, з появою обчислювальної техніки, бази знань, розташованих в книгах, переносяться на електронні носії.

Зміст видаваної інформації формується даними, зібраними в довідкових масивах системи. А пошук необхідної інформації - це сукупність різних процедур, метою яких є допомога зацікавленому знайти потрібну інформацію, тобто вибрати із загального потоку інформації необхідні дані, відповідні запиту.

Вимоги, що висуваються до інформаційно-довідкових систем є необхідність тривалого зберігання великих обсягів систематизованої інформації, яка може мати складну внутрішню структуру; також поповнення і оновлення інформації, яка повинна зберігатися та забезпечення обміну інформацією з користувачами [4].

Характерними прикладами інформаційно-довідкових систем є довідкові міські служби, бібліографічні відділи в бібліотеках.

Ще в 50-і роки XX століття люди зрозуміли, яку роль відіграє інформація як найважливіший ресурс будь-якої організації, та й суспільства в цілому. Почали розроблятися автоматизовані інформаційні системи. Спочатку це була обробка рахунків і розрахунків зарплати, які реалізовувалися на електромеханічних бухгалтерських рахункових машинах.

Але вже в 60-і роки масово стали використовувати перші електронно цифрові машини.

Це дозволило скоротити кількість обслуговуючого персоналу, значно підвищило швидкість отримання результату і як наслідок дало відчутну економію витратних матеріалів і електроенергії, дозволило підвищити якість продукції і здійснювати надійний контроль.

В даний час коло неарифметичних використання обчислювальних машин неухильно розширюється. Інтенсивно ведуться розробки в області формалізації роботи економіко-статистичного та комбінаторного властивості. Характерним прикладом є складання розкладів для залізничного, повітряного та іншого транспорту, планування виробництва і постачання, управління виробничими підприємствами та інше.

Але для всіх цих робіт необхідно створювати інформаційно-довідкові системи. Першими інформаційно-довідковими системами були електронні каталоги-картотеки, але в процесі розвитку обчислювальної техніки вони неухильно змінювалися, стаючи все більш надійними і ефективними помічниками в сучасному житті людини.

Необхідність накопичувати, аналізувати, систематизувати, коригувати та зберігати великі обсяги матеріалів і привело до розробки таких систем. Зараз інформаційно-довідкові системи швидко і легко можуть оновлювати свої бази даних, а також швидко знаходити необхідну інформацію.

Одним із серйозних завдань сьогодні, стояше перед багатьма країнами і прописаних в діючому законодавстві, є побудова інформаційного суспільства або, інакше кажучи, проведення інформатизації країни.

Електронні бази та сайти сьогодні успішно змінюють друковані книги з рецептами кулінарних страв, так як в них набагато швидше знайти необхідний рецепт, і вони не займають видимого місця в будинку. Користуватися такими джерелами набагато зручніше.

Людей цікавить не тільки сам рецепт, кількісний склад продуктів, але і час, який вони витратять на приготування, який вигляд воно повинно мати на виході і ступінь складності його приготування.

Старовинні кулінарні книги, бабусині рецепти, авторські рецепти, отримані експериментальним шляхом, які перенесені в електронну інформаційну форму, отримують іншу якість, яке дозволяє їм більш широке поширення і оперативне використання.

Основним моментом для поширення кулінарної інформації є збір і введення інформації, а також її підтримка.

Велика кількість інформаційних довідкових систем, функціонують на різних серверах і зберігають інформацію про кулінарні спільнотах, але ця інформація розкидана по інтерфейсів різних кулінарних систем, що не дає можливості робити один загальний пошук по всіх системах.

Зараз вже сформульовані певні вимоги і рекомендації для побудови будь-яких інформаційно довідкових систем, включаючи використанням Інтернет-технологій.

Створена інформаційно-довідкова система на основі таких вимог збільшує продуктивність і прискорює процес обробки даних, обслуговуюча користувача інформацією для прийняття рішень.

2.2. Вимоги до системи

Методи дослідження, які були використані для досягнення поставленої мети в магістерської дисертації:

- ідеалізація і конкретизація;
- вивчення інформаційно довідкових систем;
- узагальнення інформаційно довідкових систем;
- зіставлення;
- дедукція та індукція;
- схожість;
- угруповання;
- синтез.

Структура інформації в системі впливає на два показники: ефективність і швидкість виведення системи пошукачем і призначений для користувача фактор. Тому при розробці схеми системи необхідно щоб були задоволені вимоги як користувача, так і пошукової системи.

Нами були вирішені наступні завдання для досягнення зазначеної мети:

- інформаційно довідкової системі необхідно мати зручне поповнення бази. Для цього ми повинні розглянути існуючі стандарти, технології та протоколи в області проектування і побудови інформаційно довідкових систем з доступом через Internet і зробити порівняльний аналіз. Виходячи з результату цього аналізу розробити інформаційно довідкову систему, яка буде пов'язана з кулінарією, і буде відповідати всім таким вимогам користувачів, як інформативність, зручність знаходження рецептів, інгредієнтів, а також різних статей і новин з даної тематики і простота її використання;

- розробити продуктивний алгоритм пошуку рецептів по заданих інгредієнтів;

- створити базу даних і програмний комплекс для обслуговування такої довідкової системи;

- розглянути продуктивність створеної інформаційно-довідкової системи та розроблених алгоритмів;

- сформулювати схеми даних для розташування інформації в предметній області кулінарної галузі;

- розробити структурний план управління інформаційно-довідковою системою, і на його підставі розробити системний підхід до пошуку даних в цій системі;

- створити інтернет-портал для доступу до інформаційно-довідкової системи;

- інформаційно довідкової системі в кулінарної галузі необхідно мати можливість раціонального поповнення бази даних, а також високий рівень захисту цих даних;

- необхідно об'єднати в одному місці кулінарні інгредієнти і їх властивості та характеристики, різноманітні кулінарні рецепти, а також останні статті та новини з даної тематики в інформаційно довідковій системі в кулінарної галузі. Доступ до системи повинен бути як з комп'ютера, так і планшетних і мобільних пристроїв;
- необхідно, щоб всі рецепти мали назву, картинку, категорію, до якої належить цей рецепт, час приготування, на яку кількість осіб він розрахований, кількість білків, жирів, вуглеводів і калорій на 100 грамів продукту, список інгредієнтів, і покрокове приготування страви;
- необхідно, щоб кожна новина і стаття складалася з назви, дати публікації, титульного зображення, інформації про авторство зображення, тексту, тегів і джерела інформації;
- архітектура довідкової системи, повинна забезпечувати високу продуктивність при великій кількості призначених для користувача запитів;
- необхідно створити систему, яка реалізує надійність функціонування системи в цілому, надає надійний захист від несанкціонованого доступу, тобто забезпечить її безпеку, а також надає користувачеві зручний і зрозумілий інтерфейс.

2.3 Склад зовнішньої і внутрішньої структури системи

Проаналізувавши таблицю 1.1 отримали всі переваги і недоліки досліджуваних веб систем.

Якісна веб-система відрізняється правильно організованою структурою. Для того, щоб уникнути багатьох помилок і проблем в майбутньому необхідно планувати структуру системи до початку її реалізації.

Структура системи складається з логічної розмітки і фізичного зв'язку сторінок системи, а так само розташування видимих елементів дизайну,

визначена стандартами створення системи. Структуру розділяють на зовнішню і внутрішню.

Зовнішня структура складається з розташування видимих блоків в системі (сайтбари, шапка, інформери, футер, службові форми і інші блоки). Створення зовнішньої структури невід'ємна від технічного створення системи.

Внутрішня структура складається з приналежності матеріалів до певних категорій, а категорій до розділів, (інакше кажучи - рубрикацію), і так само кількість посилань зв'язку сторінок. Іноді рубрикацію називають логічною структурою.

2.4 Структура системи - змішана структура

Аналізуючи дані таблиці 1.1 видно, що більш затребувані категорії - це рецепти, відео цих рецептів, інформація про інгредієнти, новини і статті а так само пошук по ним, сюди додаємо тематичні анекдоти, так як відомо, що сміх продовжує життя.

Виходить такий перелік категорій:

- рецепти;
- відео рецептів;
- інформація про інгредієнти;
- статті;
- новини;
- анекдоти.

З таблиці 1.1 видно, що слушною і найбільш використовуваною структурою є деревоподібна структура.

Але при використанні деревовидної структури сайту треба постійно дивитися за її розростанням і дотримуватися золотой середини.

Для того, щоб уникнути недоліків деревовидної структури, був використаний нестандартний і мало використовуваний варіант, тобто змішану структуру.

Змішана структура системи показана на рисунку 2.1.

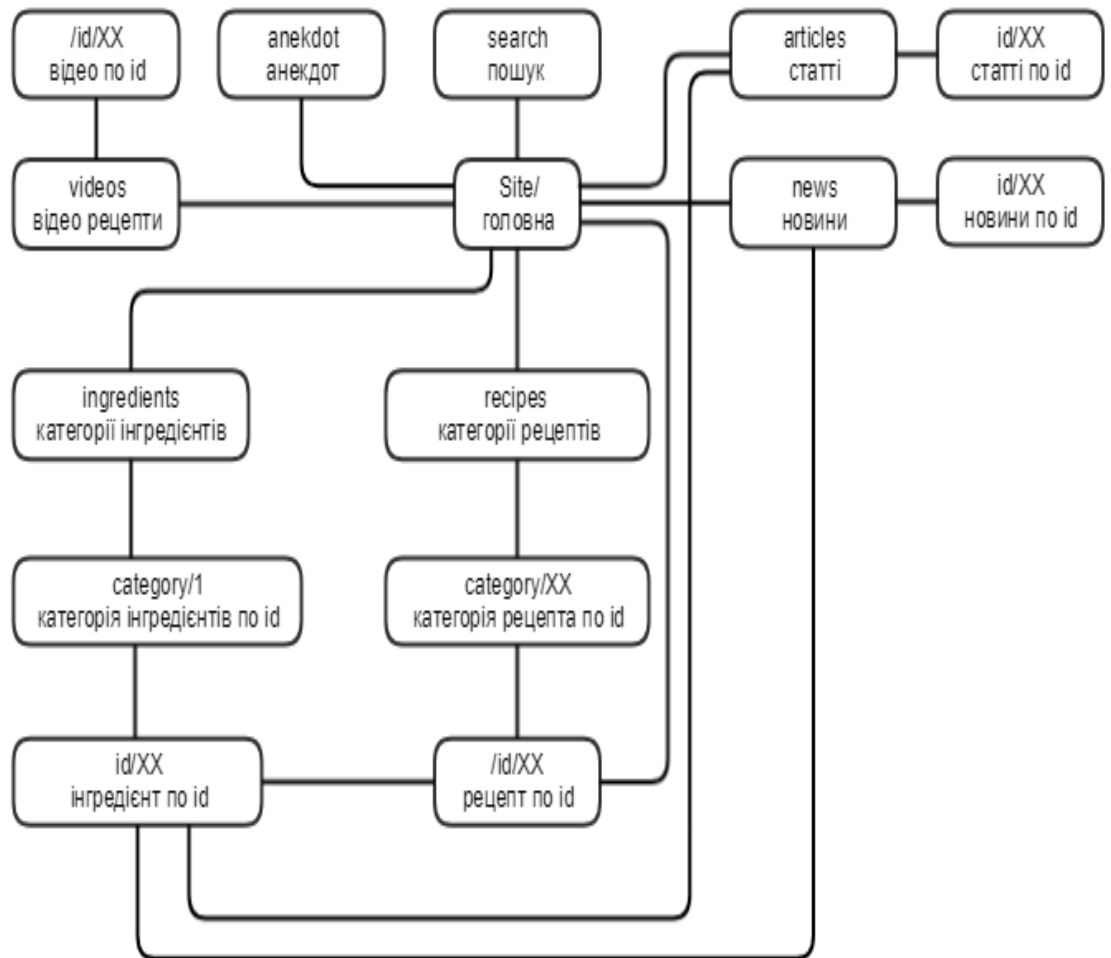


Рисунок 2.1 - Змішана структура системи

Головним завданням дизайну сайту є об'єднання всіх інформаційних блоків і створення у відвідувача приємного враження. Дизайн показує загальний стиль системи і дає відвідувачеві з першого погляду зрозуміти, що його тут чекає. Однією з головних причин, що визначають відвідуваність будь-якої веб-сторінки є дружній інтерфейс користувача.

Дизайн системи показаний на рисунку 2.2.

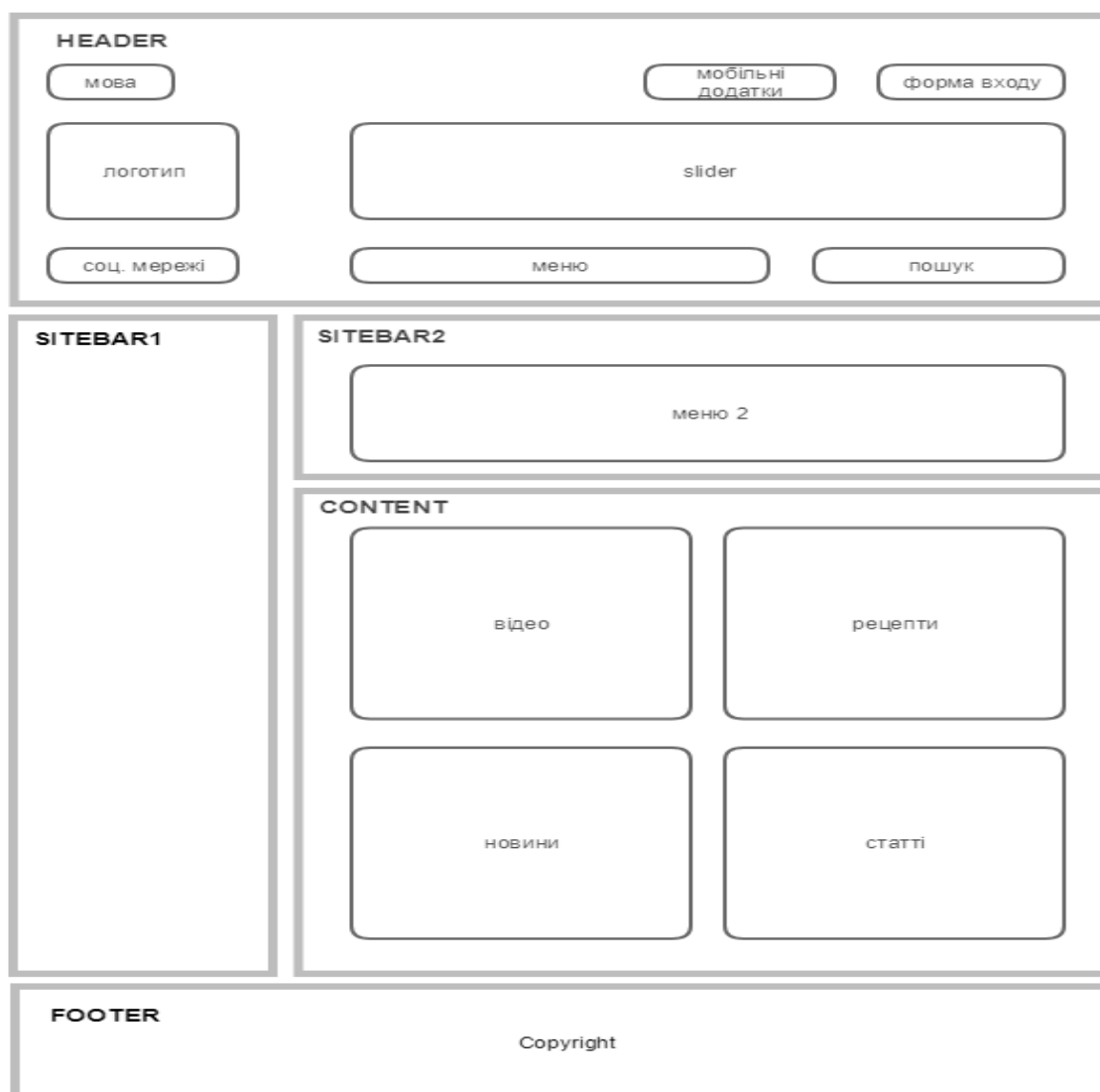


Рисунок 2.2 - Дизайн системи

Після визначення зовнішнього вигляду головної сторінки сайту, необхідно було обдумати і виробити шаблон внутрішніх сторінок сайту, який був зроблений однаковим для всіх розділів і розділений на чотири типи. Так користувачеві зручніше працювати а сайтом. Перший тип - для розділу з категоріями, другий - для розділу зі списком рецептів, третій - для розділу зі списком новин і статей, четвертий - для утримання конкретної сторінки.

Дизайн розділу категорій для рецептів і інгредієнтів показано на рисунку 2.3.

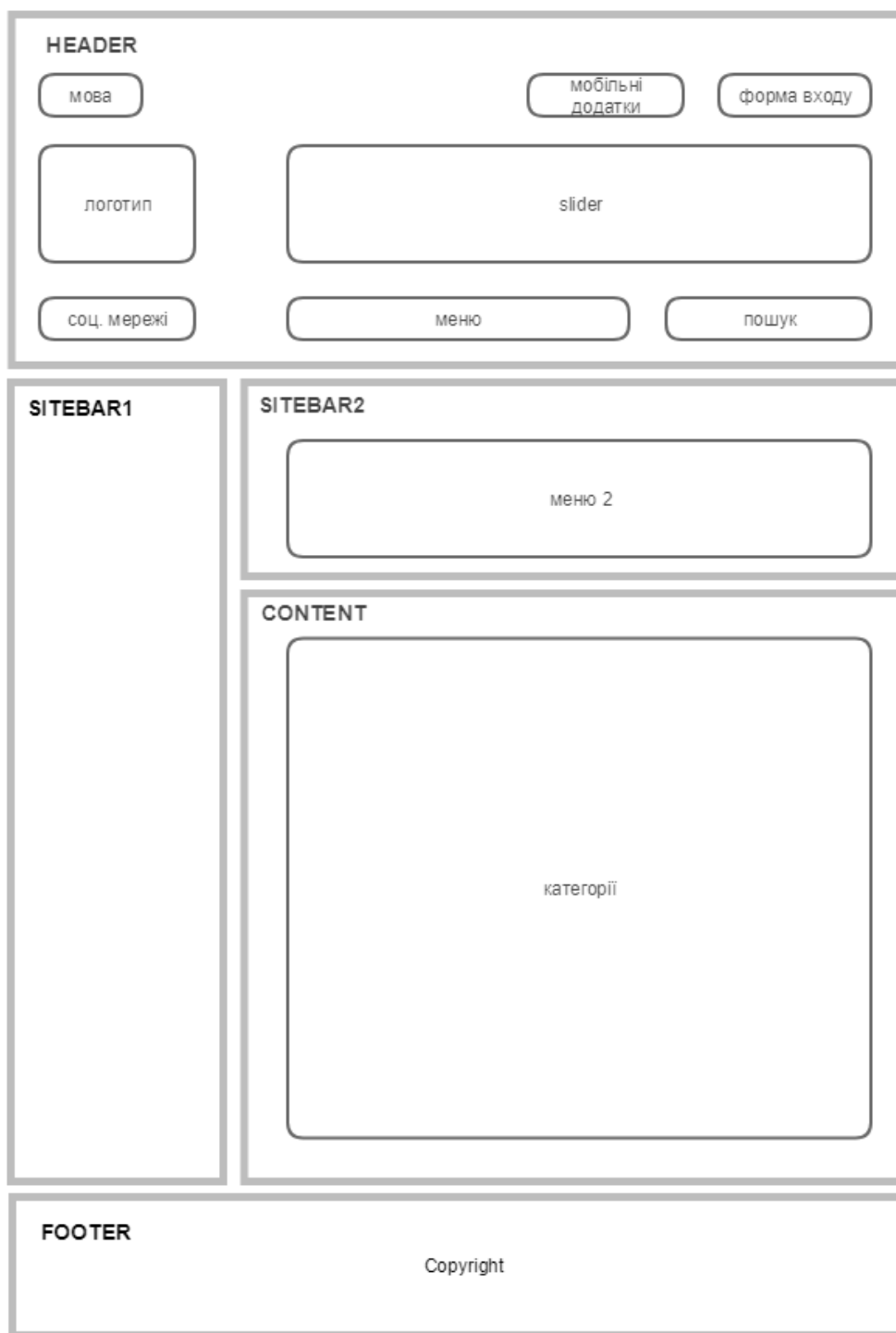


Рисунок 2.3 - Дизайн розділу категорій

Для розділу категорії рецептів і розділу категорій інгредієнтів був використаний окремий шаблон.

Зовнішній вигляд розділу списку рецептів показаний на рисунку 2.4.

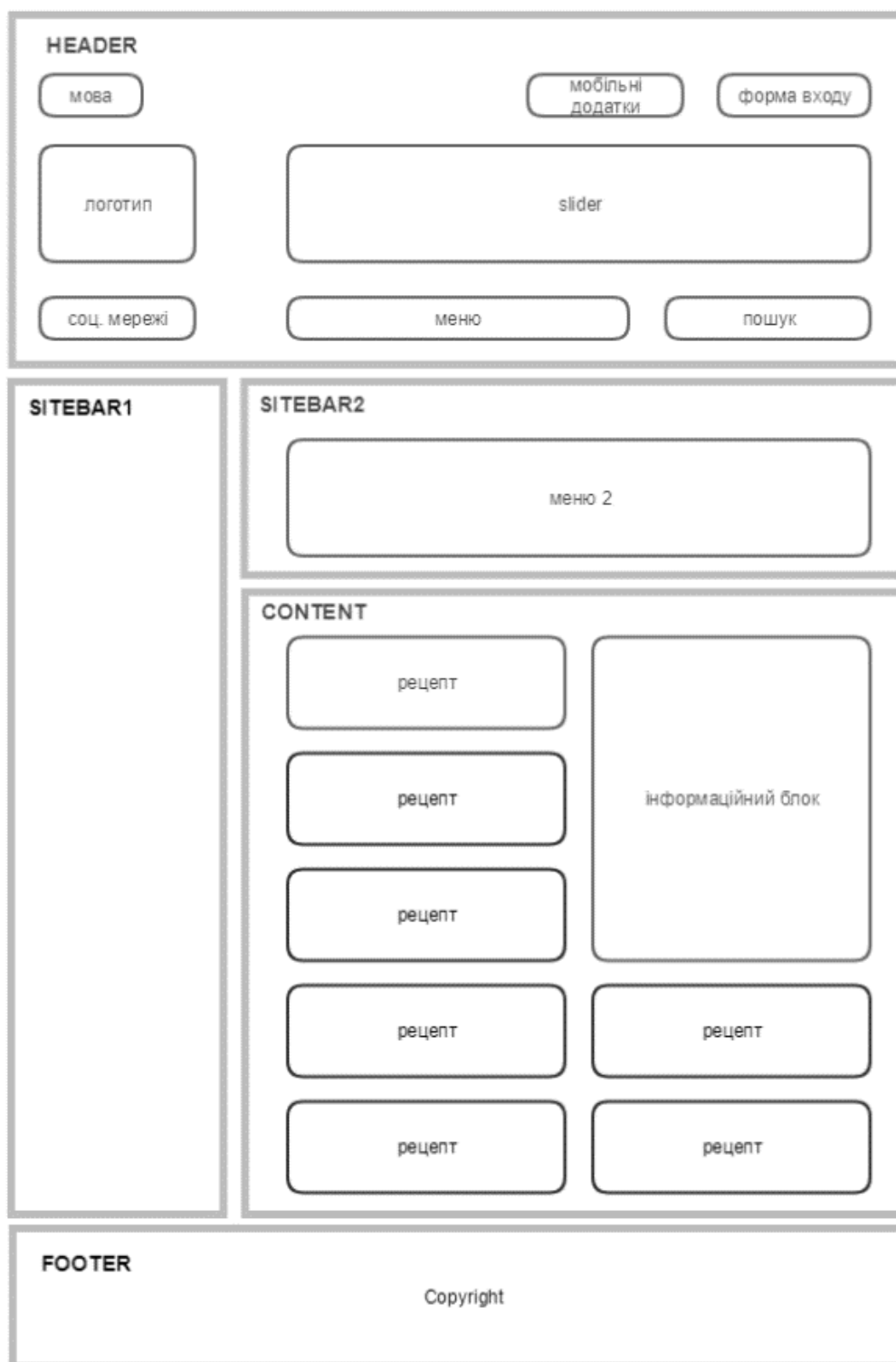


Рисунок 2.4 - Зовнішній вигляд розділу списку рецептів

Зовнішній вигляд розділу списку новин/статей показаний на рисунку 2.5.

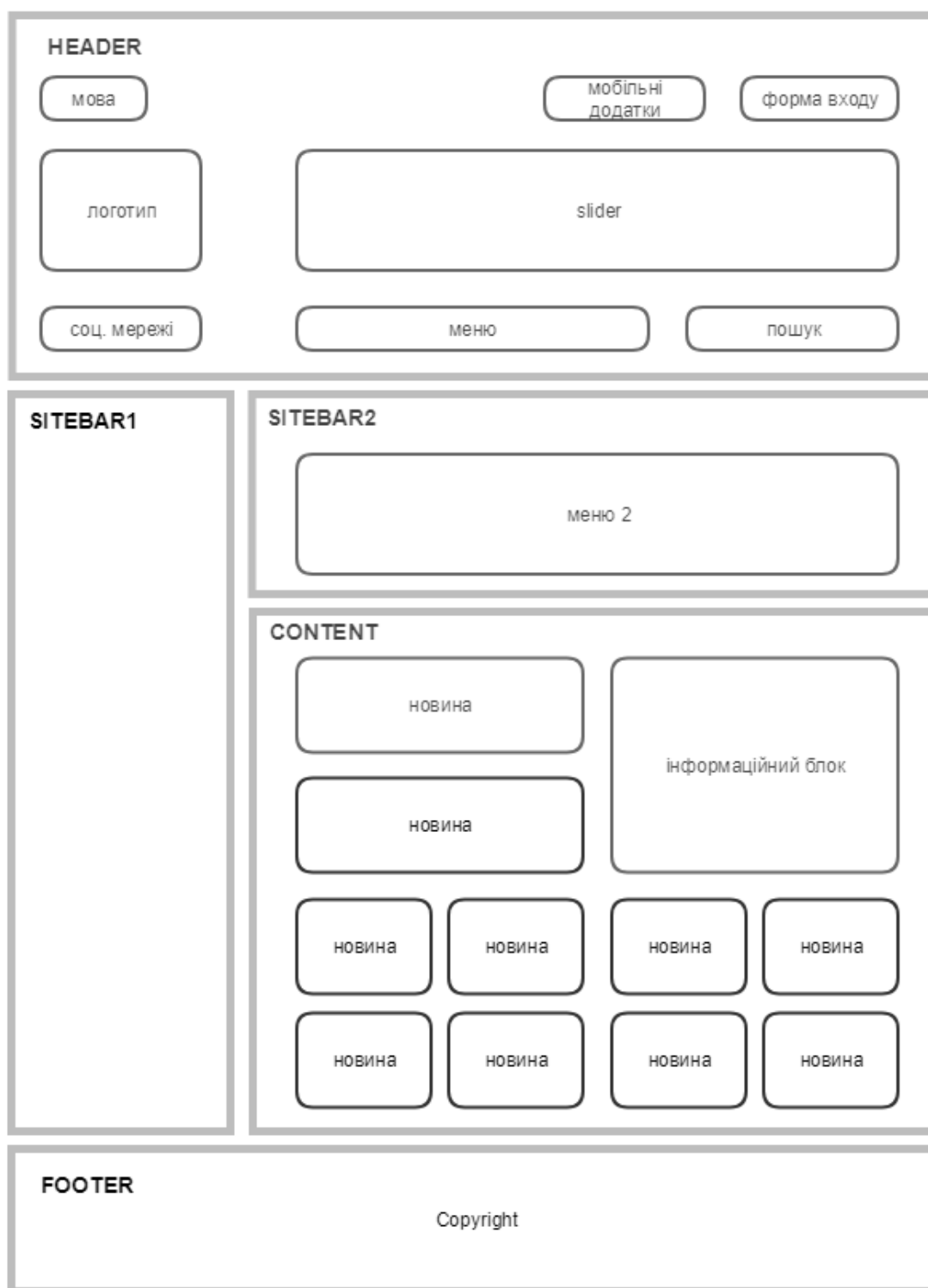


Рисунок 2.5 -План розділу списку новин/статей

Розміщення інформації конкретного рецепту, інгредієнта, статті, новини або відео показано на рисунку 2.6.

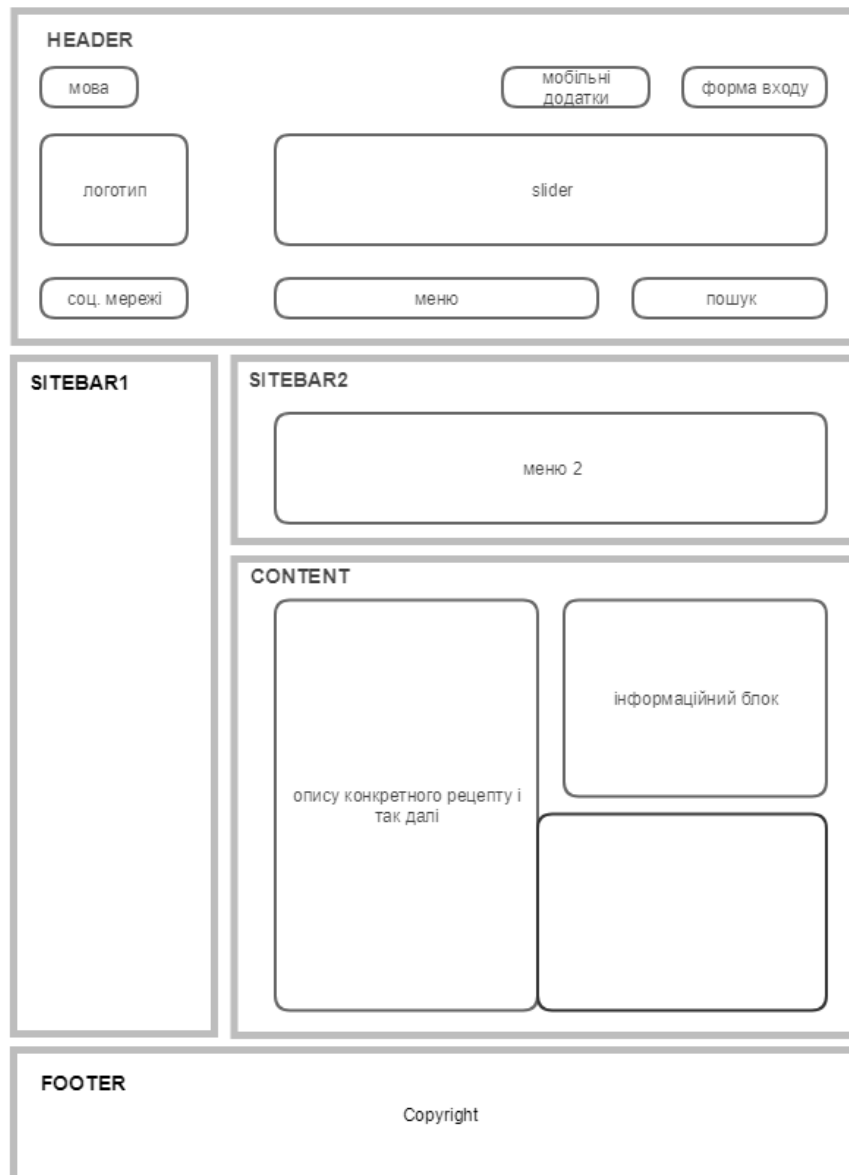


Рисунок 2.6 - План розташування інформації

2.5 Вимоги до розділів системи

З рисунка 2.2 видно, що на системі повинні бути блоки header, sitebar1, sitebar2, content і footer, причому для всіх сторінок систему інформація у всіх блоках, виключаючи sitebar2 і content повинна бути однаковою.

Header включає в себе перемикач мови систему, форми входу, логотипу, слайдера, посилання на мобільні додатки, посилання на соціальні мережі, меню і пошуку.

Sitebar1 включає в себе додаткову інформацію про систему.

Sitebar2 включає в себе 3 види перемикачів міні меню для рецептів і інгредієнтів відповідно для кожного розділу.

Головний блок Content, змінюється відповідно від розділу, в якому знаходиться. Нижче всі описи розділів стосуватиметься тільки цього блоку.

Footer складний з додатковою інформацією про систем.

Головна сторінка

З рисунка 2.2 видно, що блок content розділений на чотири підблока для рецептів, відео, новин і статей.

У блоків рецептів і відео однакова структура, яку видно на рисунку 2.7.

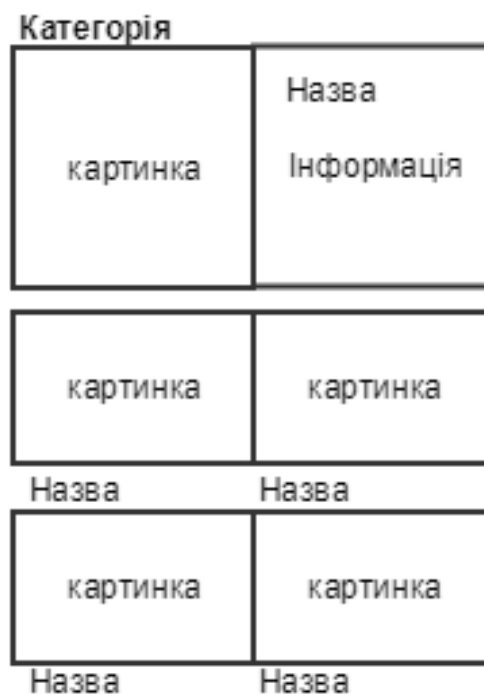


Рисунок 2.7 - Структура блоку відео та рецептів

У свою чергу блоки новин і статей теж мають однакову структуру, що показано на рисунку 2.8.

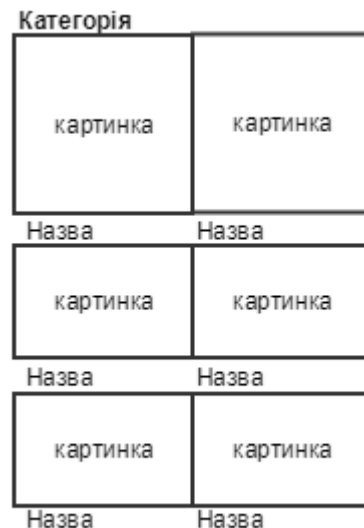


Рисунок 2.8 - Структура блоку новини та статі

Розділи категорій рецептів та інгредієнтів

З рисунка 2.3 видно, що розділи складаються тільки зі списку категорій.

Розділ конкретної категорії рецептів. З рисунка 2.4 видно, що розділ категорії рецептів включає в себе список рецептів, в якому будь-який рецепт складається з назви, картинки, і короткої інформації про цей рецепт.

Розділ конкретної категорії інгредієнтів включає в себе список інгредієнтів, їх назви, картинки інгредієнтів, Б / Ж / на 100 грам і короткої інформації про кожного.

Розділи новин і статей включають в себе список новин або статей відповідно розділу, що видно з рисунка 2.5.

Інформація про рецепт включає в себе назву рецепта, на яку кількість осіб розрахований, часу приготування, б / Ж / на 100 грам, картинки страви, інгредієнтів, пояснення, для чого використовуються ці інгредієнти і покроковий опис приготування з картинками і часом на кожен виконуваний крок.

Відео інформація про рецепт включає назву рецепту, відео, список інгредієнтів і покрокове приготування страви.

Інформація про інгредієнт включає його назву, картинку інгредієнта, Б / Ж / на 100 грам і повну інформацію про нього.

Стаття/Новина включає картинку, назву, інформацію про авторство картинки, безпосередньо текст самої новини/статті, теги та джерела новини/статті.

2.6 Висновок до розділу

Аналізуючи типи застосовуваних системних структур і беручи до уваги оцінку внутрішніх взаємозв'язків системи була обрана змішана структура системи, як оптимально слухна структура. Також був розроблений склад внутрішньої і зовнішньої структури системи. Разом з тим, були розроблені вимоги до розділів.

3 КОНСТРУЮВАННЯ БАЗИ ДАНИХ

3.1 Види систем управління базами даних

База даних (БД) - іменована сукупність даних, переважно великих обсягів, яка організована за певними правилами, що передбачає загальні принципи опису, зберігання, дії над цими даними, незалежними від прикладних програм. Динамічні сайти зі значними обсягами даних активно використовують бази даних. Це можуть бути інтернет-магазини, корпоративні сайти, портали. [17]

Система керування базами даних (СКБД) - сукупність програм, призначених для управління БД, а також ймовірності отримання користувачами необхідної інформації з бази [17].

Існує чотири структурних типу систем управління базами даних:

- реляційні бази даних;
- ієрархічні бази даних;
- мережеві бази даних;
- об'єктно-орієнтовані бази даних.

Основними завданнями СКБД є:

- 1) формування та підтримка БД;
- 2) гарантія цілісності БД;
- 3) прийом запитів;
- 4) пошук і обробка інформації;
- 5) надання інформації користувачам;
- 6) організація спільної роботи користувачів.

Ієрархічна модель даних - це модель даних, де бази даних представлені у вигляді дерева (ієрархічної структури), яка складається з об'єктів (даних) різних рівнів.

Всі об'єкти пов'язані між собою і будь-який об'єкт може включати в себе кілька об'єктів нижчого рівня. Ці об'єкти складаються щодо предка (об'єкт більш близький до кореня) до нащадка (об'єкт більш низького рівня), при цьому об'єкт-предок може не мати нащадків або має їх кілька, в той час об'єкт-

нащадок обов'язково має тільки один предка. Об'єкти, які мають спільного предка, називаються близнюками (в програмуванні щодо структури даних дерево усталене назва брати).

Логічна модель даних, яка є розширенням ієрархічного підходу, точна математична теорія, яка описує структурну сторону, сторону цілісності і сторону обробки даних в мережевих базах даних називається мережевою моделлю даних.

Ієрархічна модель даних відрізняється від мережевої тим, що в ієрархічних структурах запис-нащадок повинна мати тільки одного предка, а в мережевій структурі даних нащадок може мати будь-яку кількість предків.

Зображення мережевий СУБД з графічним представленням зв'язків показано на рисунку 3.1 [5].

Network Model

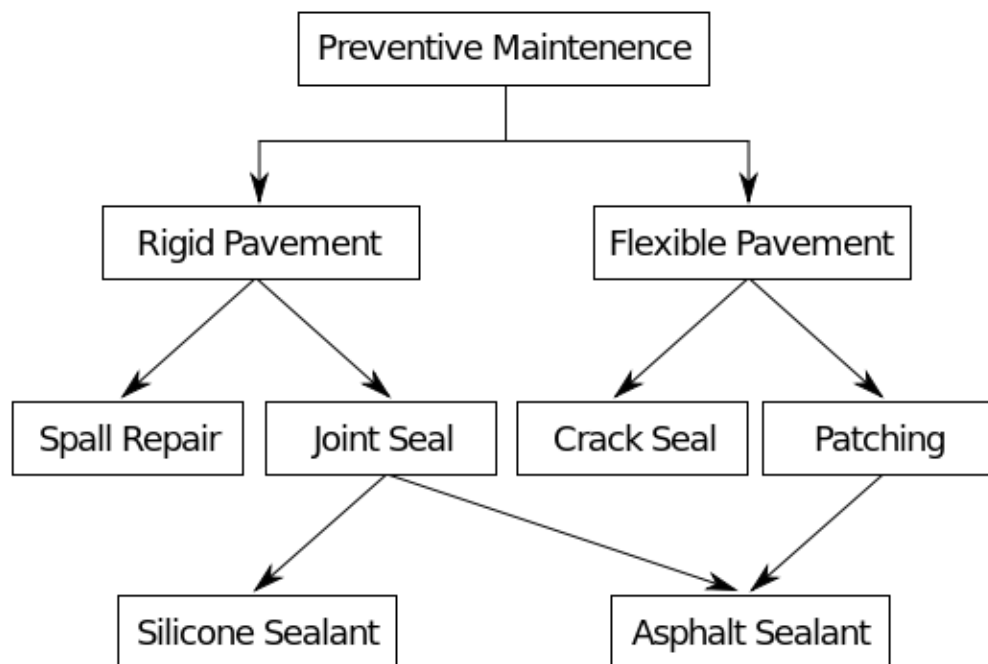


Рисунок 3.1 - Мережева модель даних

Зв'язок між файлами даних в реляційних базах даних є реляційний. Користувач повинен передавати ієрархію для доступу до необхідних даних в ієрархічних і мережевих базах даних. Використовуючи загальні номери даних

або ключове поле ці бази даних підключаються до даних в різних файлах. У реляційних базах дані зберігаються в різних таблицях управління доступом, де кожен має ключове поле, яке ідентифікує кожен рядок.

Об'єктно-орієнтована база даних (ООБД) це база даних, де дані моделюються у вигляді об'єктів, їх атрибутів, методів і класів. Об'єктно-орієнтовані бази даних рекомендують для випадків, коли необхідна високопродуктивна обробка даних, які мають складну структуру.

3.2 MySQL -вільна реляційна система управління базами даних

Для реалізації системи базою даних була обрана MySQL через своїх переваг перед іншими базами даних.

Основними достоїнствами MySQL є:

- відкритість коду;
- спільнота;
- системи;
- безпека;
- швидкодія;
- надійність;
- ліцензія;
- переносимість.

Швидкодія MySQL досить висока, через внутрішнього механізму багатопоточності.

Завдяки тому, що в базі даних MySQL можна налаштовувати який користувач, з якого домена і з якою таблицею має право працювати і які команди може застосовувати забезпечується високий рівень безпеки.

Користувачі, які мають доступ до БД `darkside_kitchen` показані на рисунку 3.2.




База данных	Размер	Привилегированные пользователи
darkside_kitchen	51.03 MB	darkside_admin  darkside_fk_site  darkside_mob 

Рисунок 3.2 – Користувачі, котрі мають доступ до БД darkside_kitchen

MySQL розповсюджується безкоштовно для некомерційних цілей.

Ви можете додати власну необхідні вам функції в пакет і розширити його функціональність так, як вам потрібно, тому що код є відкритим.

Щоб база даних працювала стабільно і на неї не впливали зовнішні фактори розробники MySQL передбачили всі фактори.

Її можна розгорнути на будь-якому десктопном пристрою, тому що MySQL, не надто вимоглива до ресурсів системи.

Завдяки відкритості коду існує багато майданчиків, де розробники обмінюються досвідом і діляться порадами, що спрощує пошук рішення частини проблем.

Завдяки тому, що MySQL можна встановити практично на будь-яку доступну платформу, це робить роботу з нею ще більш легкої [6].

3.3 Створення бази даних в MySQL

Для того, щоб створити бази даних зайшли в MySQL через cPanel хостингу. База даних носить назву darkside_kitchen.

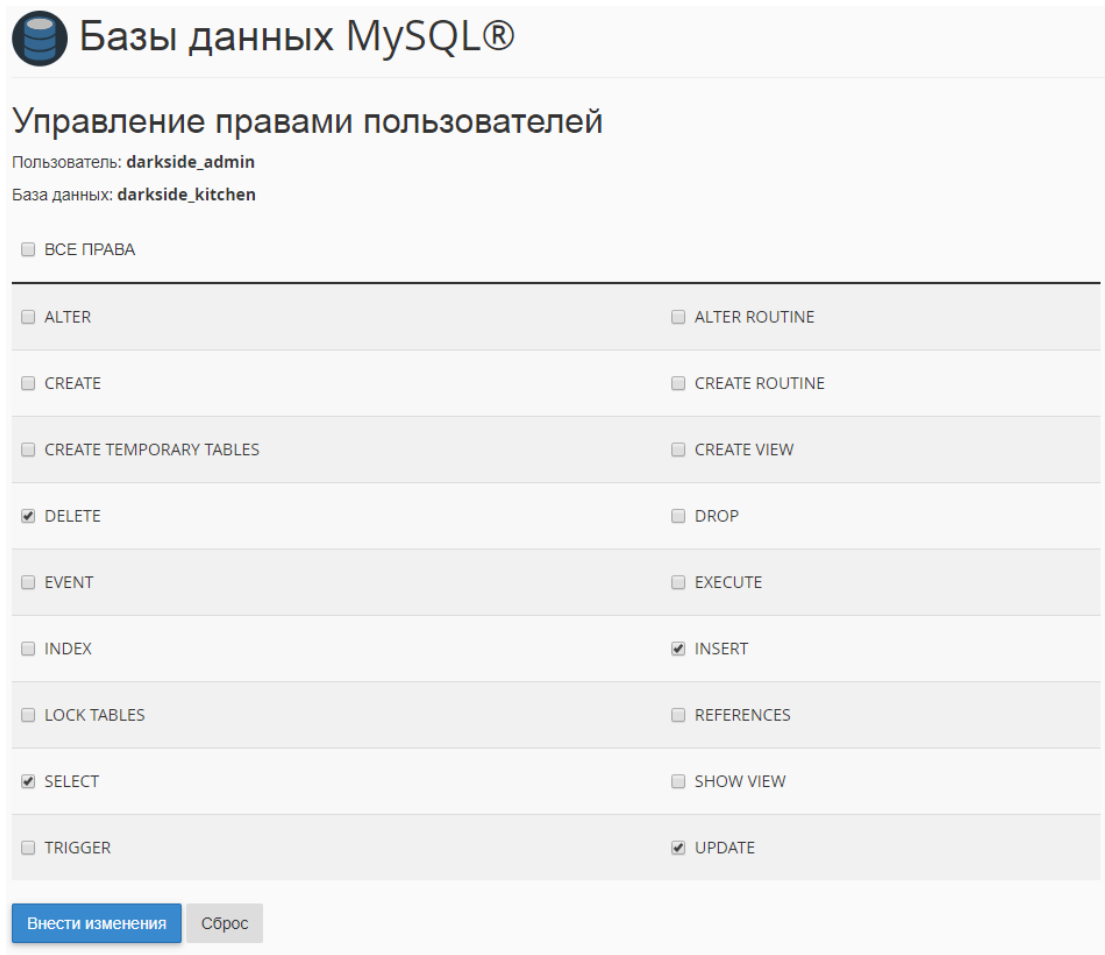
Необхідно було створити користувачів для роботи з нею. Було створено трьох користувачів з різним рівнем доступу до бази даних для підвищення безпеки і контролю дій, пов'язаних з базою даних.

Наші користувачі бази даних:

- darkside_admin;
- darkside_fk_site;
- darkside_mob.

darkside_admin - користувач має найбільший набір прав впливу на базу даних.

Права користувача бази даних darkside_адмін показані на рисунку 3.3.



Базы данных MySQL®

Управление правами пользователей

Пользователь: **darkside_admin**

База данных: **darkside_kitchen**

☐ ВСЕ ПРАВА

<input type="checkbox"/> ALTER	<input type="checkbox"/> ALTER ROUTINE
<input type="checkbox"/> CREATE	<input type="checkbox"/> CREATE ROUTINE
<input type="checkbox"/> CREATE TEMPORARY TABLES	<input type="checkbox"/> CREATE VIEW
<input checked="" type="checkbox"/> DELETE	<input type="checkbox"/> DROP
<input type="checkbox"/> EVENT	<input type="checkbox"/> EXECUTE
<input type="checkbox"/> INDEX	<input checked="" type="checkbox"/> INSERT
<input type="checkbox"/> LOCK TABLES	<input type="checkbox"/> REFERENCES
<input checked="" type="checkbox"/> SELECT	<input type="checkbox"/> SHOW VIEW
<input type="checkbox"/> TRIGGER	<input checked="" type="checkbox"/> UPDATE

Рисунок 3.3 – Права користувача darkside_admin

З рисунка 3.3 видно, що користувач в праві використовувати команди DELETE, SELECT, INSERT і UPDATE. Будь-які інші команди доступні тільки на пряму через інтерфейс MySQL для супер адміністратора.

Користувачам darkside_fk_site і darkside_mob доступні тільки команди SELECT, INSERT і UPDATE, хоча перший - це користувач для роботи з сайтом, а другий - для роботи з API системою.

Щоб працювати власне з створеної базою даних була використана програма PMA.

3.4 Структура таблиць і їх внутрішній зміст

Типи даних, які використовуються в MySQL відображені в таблиці 3.1

Таблиця 3.1 – Типи даних

Тип	Опис
int	INT є синонімом для INTEGER
varchar	VARCHAR є рядки змінної довжини
float	Призначений для подання дійсних чисел одинарної точності
timestamp	Тип стовпчика забезпечує тип представлення даних, який можна використовувати для автоматичного запису поточних дати
text	Зберігаються, як окремі об'єкти з таблиць та наборів результатів, що містять їх

Таблиця БД text_articles, яка відповідає за новини і статті показана в таблиці 3.2.

Таблиця 3.2 – Таблиця БД text_articles

Поле	Тип(Довжина)	Опис
text_id (Primary Key) AUTO_INCREMENT	int(10)	id новини або статті
type_status	tinyint(2)	Тип новина або стаття
text_status	tinyint(2)	Черновик або Опубліковано
name_ru	text	Заголовок (російська мова)
name_uk	text	Заголовок (українська мова)
name_en	text	Заголовок (англійська мова)
Image_name	text	Назва картинки
text_ru	text	Текст (російська мова)
text_uk	text	Текст (українська мова)

Продовження таблиці 3.2

Поле	Тип(Довжина)	Опис
text_en	text	Текст (англійська мова)
last_update ON UPDATE CURRENT_TIMESTAMP	timestamp	Дата створення
published	timestamp	Дата публікації
show_title_image	tinyint(2)	Зображення показувати/сховувати
text_author	text	Автор матеріалу
link_source	text	Походження джерела
name_source	text	Назва джерела

Зв'язок таблиць БД пов'язаних з новинами і статтями показана на рисунку 3.4.

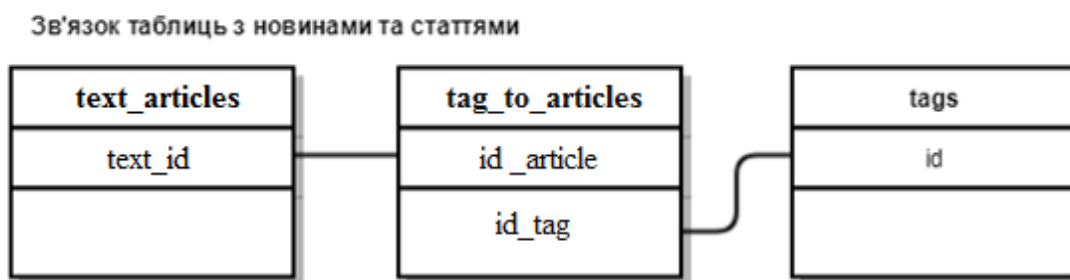


Рисунок 3.4 – Зв'язок таблиць БД

Таблиця 3.2 містить всі новини і статті системи.

У таблиці 3.3 показана таблиця `tag_to_articles`, яка містить `id` статей/новин і `id` тега до них.

Таблиця 3.3 - Таблиця БД `tag_to_articles`

Поле	Тип(Довжина)	Опис
id_article	int(9)	id новини або статті
id_tag	int(9)	id тегу

У таблиці 3.4 показана таблиця БД site_bookmarks, яка відповідає за закладки користувачів системи.

Таблиця 3.4 – Таблиця БД site_bookmarks

Поле	Тип(Довжина)	Опис
id_type (Primary Key)	varchar(254)	Характеристика запису: відео, інгредієнт, рецепт, новина, стаття, або анекдот
id_target (Primary Key)	int(9)	id типу запису
id_account (Primary Key)	int(9)	id користувача
status	tinyint(2)	Статус запису (прибраний / доданий)

Таблиця 3.4 зберігає інформацію, яку користувач додав в обране в системі.

У таблиці 3.5 показана таблиця БД ingredients_category, яка відповідає за категорії інгредієнтів.

Таблиця 3.5 – Таблиця БД ingredients_category

Поле	Тип(Довжина)	Опис
cat_ing_id (Primary Key) AUTO_INCREMENT	tinyint(2)	id категорії
ci_name_ru	varchar(45)	Назва (російська мова)
ci_name_uk	varchar(45)	Назва (українська мова)
ci_name_en	varchar(45)	Назва (англійська мова)

У таблиці 3.5 міститься назва категорії інгредієнтів на трьох мовах.

Таблиця БД recipe_category показана в таблиці 3.6, що відповідає за категорії рецептів.

Таблиця 3.6 – Таблиця БД recipe_category

Поле	Тип(Довжина)	Опис
ca_r_id (Primary Key) AUTO_INCREMENT	tinyint(3)	id
category	tinyint(3)	id категорії вищого рівня
name_category_ru	varchar(65)	Назва (російська мова)
name_category_uk	varchar(65)	Назва (українська мова)
name_category_en	varchar(65)	Назва (англійська мова)

У таблиці 3.6 назву категорії рецептів представлено у трьох мовах, а також зазначена приналежність категорії до категорії вищого рівня.

Таблиця БД site_favorites показана в таблиці 3.7, яка відповідає за закладки в мобільному додатку.

Таблиця 3.7 – Таблиця БД site_favorites

Поле	Тип(Довжина)	Опис
u_id (index)	int(12)	id запис користувача
Id_recipe (index)	int(12)	id рецепта
f_status (index)	tinyint(2)	Статус запису (доданий/прибраний з обраного)
app_inh	enum(list)	Додаток до якого відноситися запис

Таблиця 3.7 містить інформацію, яку користувач додав в обране в конкретному додатку.

Таблиця БД diary_food показана в таблиці 3.8, яка відповідає за щоденник харчування і є частиною калоризатора.

Таблиця 3.8 – Таблиця БД diary_food

Поле	Тип(Довжина)	Опис
u_id (index)	int(12)	id запис користувача
df_day (index)	int(12)	Дата запису

Продовження таблиці 3.8

Поле	Тип(Довжина)	Опис
id_ingredient (index)	smallint(5)	id інгредієнта
id_recipe (index)	smallint(5)	id рецепта
gram	mediumint(6)	Число грамів
calories	mediumint(6)	Число калорій
df_status	tinyint(2)	Статус запису (доданий/прибраний з щоденника)

Таблиця 3.8 містить список рецептів і інгредієнтів, які користувач вказав на конкретний день.

Таблиця БД history_food показана в таблиці 3.9, що відповідає за історію щоденника харчування і є частиною калоризатора.

Таблиця 3.9 – Таблиця БД history_food

Поле	Тип(Довжина)	Опис
u_id (index)	int(12)	id запис користувача
hf_day (index)	int(11)	Дата запису
weight_your	smallint(4)	Вага користувача
weight_food	smallint(4)	Вага з'їденої їжі
calories	mediumint(5)	Кількість калорій
water	mediumint(5)	Кількість випитої води

Історія вживання їжі і води користувачем по днях відображається в таблиці 3.9.

Таблиця БД info_food показана в таблиці 3.10, яка зберігає інформацію про користувача и є частиною калоризатора.

Таблиця 3.10 – Таблиця БД info_food

Поле	Тип(Довжина)	Опис
u_id (Index)	int(12)	id запис користувача
sex	tinyint(2)	Стать (чоловіча або жіноча)
weight	tinyint(4)	Дані користувача (вага)
age	tinyint(3)	Дані користувача (вік)

Продовження таблиці 3.10

Поле	Тип(Довжина)	Опис
growth	tinyint(4)	Дані користувача (зріст)
activity	tinyint(2)	Дані користувача (активність)
status_diet	tinyint(2)	Тип дієти

На рисунку 3.5 показано зв'язок таблиць БД пов'язаних з калоризатором.

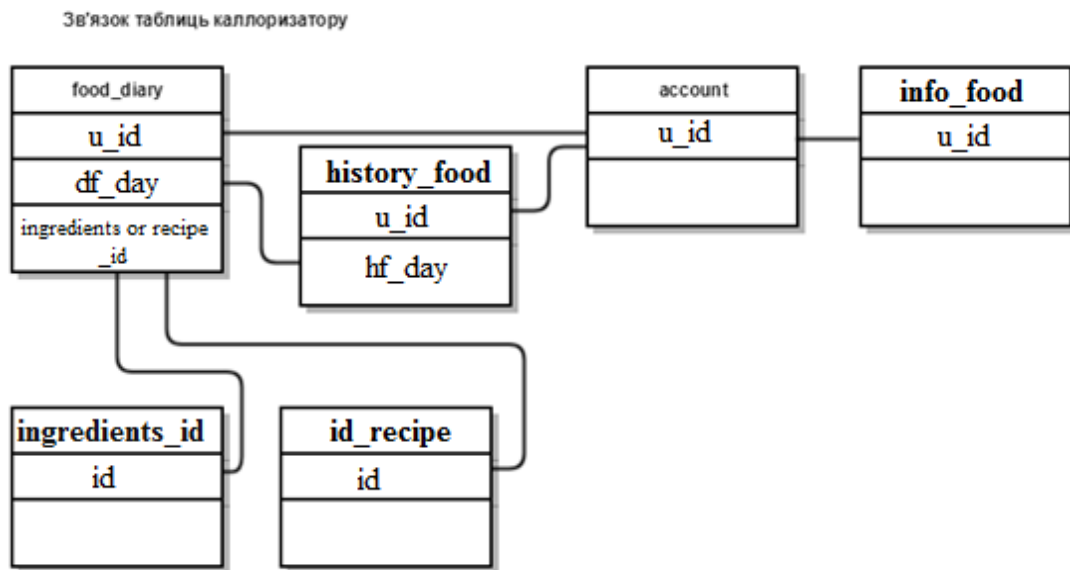


Рисунок 3.5 – Зв'язок таблиць БД

Таблиця 3.10 застосовується для розрахунку дієти користувача, виходячи з запроваджених ним даних.

Таблиця БД ingredients відображена в таблиці 3.11, що відповідає за інгредієнти.

Таблиця 3.11 – Таблиця БД ingredients

Поле	Тип(Довжина)	Опис
ingredients_id (Primary Key) AU TO_INCREMENT	int(8)	id інгредієнта
name_ingredient_ru	varchar(65)	Назва (російська мова)
name_ingredient_uk	varchar(65)	Назва (українська мова)

Продовження таблиці 3.11

Поле	Тип(Довжина)	Опис
name_ingredient_en	varchar(65)	Назва (англійська мова)
category	tinyint(3)	іd категорії до якої належить інгредієнт
image	varchar(405)	Зображення інгредієнту
protein	int(5)	Кількість білків на 100 грамів
fats	int(6)	Кількість жирів на 100 грамів
carbohydrates	int(5)	Кількість вуглеводів на 100 грамів
calories	int(6)	Кількість калорій на 100 грамів
text_ru	text	Опис (російською мовою)
text_uk	text	Опис (українською мовою)
text_en	text	Опис (англійською мовою)

Рисунок 3.6 показує зв'язок таблиць БД, які пов'язані з інгредієнтом

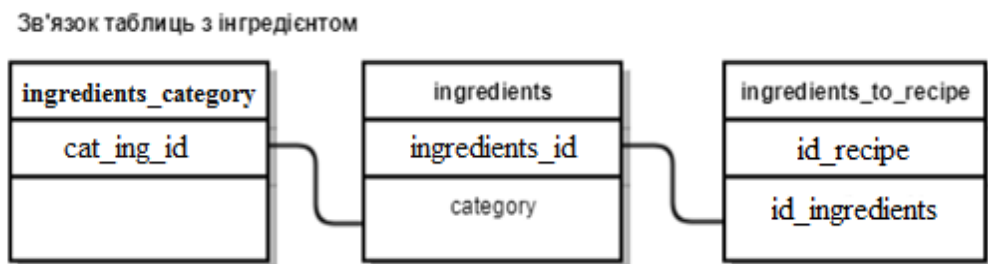


Рисунок 3.6 – Зв'язок таблиць БД, які пов'язані з інгредієнтом

Таблиця 3.11 містить в собі інформацію про всі інгредієнти, що знаходяться в системі.

Таблиця БД `ingredients_to_recipe`, відображена в таблиці 3.12, яка містить інгредієнти, які використовуються в рецепті.

Таблиця 3.12 – Таблиця БД `ingredients_to_recipe`

Поле	Тип(Довжина)	Опис
id_recipe	int(8)	іd рецепта до якого належить інгредієнт
id_ingredients	int(8)	іd інгредієнта

Продовження таблиці 3.12

Поле	Тип(Довжина)	Опис
quantity_ru	varchar(36)	Кількість інгредієнту (російська мова)
quantity_uk	varchar(36)	Кількість інгредієнту (українська мова)
quantity_en	varchar(36)	Кількість інгредієнту (англійська мова)
gram	mediumint(6)	Кількість грамів
ctg	int(3)	Категорія до якої належить інгредієнт у рецепті

Таблиця 3.12 містить інформацію про всі інгредієнти, які входять в рецепти.

Таблиця БД admin_logs показана в таблиці 3.13, в якій зберігаються всі дії адміністраторів системи.

Таблиця 3.13 – Таблиця БД admin_logs

Поле	Тип(Довжина)	Опис
logs_id (Primary Key) AUTO_INCREMENT	int(11)	id запису
admin_name	varchar(15)	Ім'я адміністратора
date ON UPDATE CURRENT_TIMESTAMP	timestamp	Дата події
action	varchar(25)	Характер дії (вхід/вихід, додавання/зміна або видалення)
info	text	Подія та id
ip	varchar(11)	IP адреса с котрої була виконана подія

Таблиця 3.13 містить інформацію про всі дії адміністраторів системи за весь час.

Таблиця БД account_admin показана в таблиці 3.14, що відповідає за права доступу адміністраторів.

Таблиця 3.14 – Таблиця БД account_admin

Поле	Тип(Довжина)	Опис
admin_name (index)	varchar(25)	Ім'я адміністратора
password	varchar(25)	Пароль адміністратора
added CURRENT_TIMESTAMP	timestamp	Дата створення облікового запису
status	tinyint(2)	Статус (активний/ні)
recipe	tinyint(2)	Право редагувати рецепти
anekdot	tinyint(2)	Право редагувати анекдоти
logs	tinyint(2)	Право дивитись логи
support	tinyint(2)	Право редагувати підтримку
statistics	tinyint(2)	Право бачити статистику
con_db	tinyint(2)	Право зберігати БД
work	tinyint(2)	Право бачити звіт про роботу інших адміністраторів
ingredients	tinyint(2)	Право редагувати інгредієнти
articles	tinyint(2)	Право редагувати новини та статті
videos	int(12)	Право редагувати відео рецепти
tags	tinyint(2)	Право редагувати теги

Таблиця 3.14 містить в собі відомість про всіх адміністраторів системи і їхні права доступу.

Таблиця БД anekdot показана в таблиці 3.15, в якій знаходяться анекдоти.

Таблиця 3.15 – Таблиця БД anekdot

Поле	Тип(Довжина)	Опис
a_id (Primary Key) AUTO_INCREMENT	int(11)	id анекдота
anekdot_ru	text	Анекдот (російська мова)
anekdot_uk	text	Анекдот (українська мова)
anekdot_en	text	Анекдот (англійська мова)
deleted	tinyint(2)	Статус (видалений/ні)

Всі анекдоти, які потрапляють в систему зберігаються в таблиці 3.15.

Таблиця БД recipe, показана в таблиці 3.16, яке береже інформацію про рецепти.

Таблиця 3.16 – Таблиця БД recipe

Поле	Тип(Довжина)	Опис
id (Primary Key Index) AUTO_INCREMENT	int(11)	id рецепта
name_recipe_ru	varchar(254)	Назва рецепту (російська мова)
name_recipe_uk	varchar(254)	Назва рецепту (українська мова)
name_recipe_en	varchar(254)	Назва рецепту (англійська мова)
time_cooking	int(6)	Час приготування страви
servings	varchar(11)	Кількість персон
protein	int(5)	Кількість білків на 100 грамів
fats	int(6)	Кількість жирів на 100 грамів
carbohydrates	int(5)	Кількість вуглеводів на 100 грамів
calories	int(6)	Кількість калорій на 100 грамів
like_r	tinyint(2)	Кількість лайків на рецепті
vision_r	tinyint(2)	Статус видно чи ні
fridge	tinyint(2)	Чи використовується холодильник
grill	tinyint(2)	Чи використовується гриль
microwave	tinyint(2)	Чи використовується мікрохвильова піч
multivarka	tinyint(2)	Чи використовується мультиварка
oven	tinyint(2)	Чи використовується піч
pan	tinyint(2)	Чи використовується каструля

На рисунку 3.7 відображено зв'язок таблиць БД, які пов'язані з рецептом.

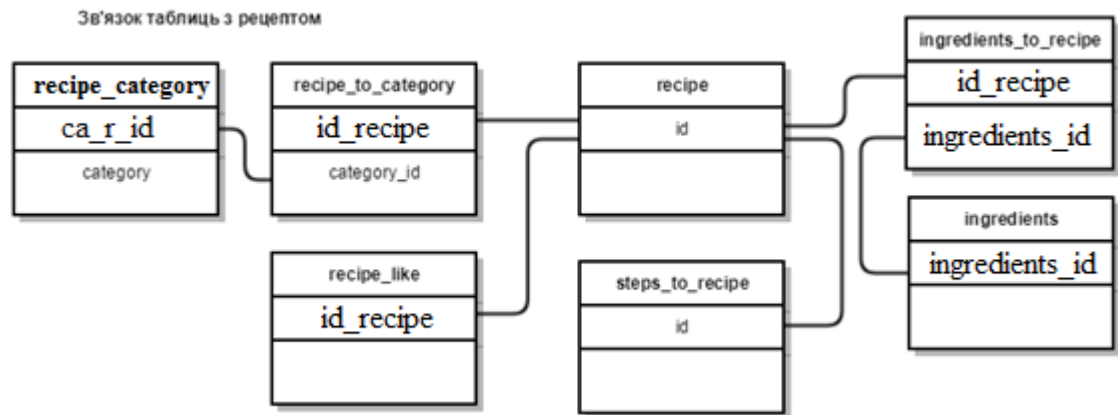


Рисунок 3.7 – Зв'язок таблиць БД, які пов'язані з рецептом

Інформація про усіх рецептах системи знаходиться в таблиці 3.16.

Таблиця БД like_recipe відображена в таблиці 3.17, яка містить інформацію про лайки в рецептах.

Таблиця 3.17 – Таблиця БД like_recipe

Поле	Тип(Довжина)	Опис
id_recipe (index)	int(7)	id рецепта
id_account (index)	int(7)	id запис користувача
vision	tinyint(2)	Відображати/ні

Таблиця 3.17 містить в собі інформацію про всі лайки до рецептів.

Таблиця БД recipe_to_category, показана в таблиці 3.18, яка містить інформацію про належність рецептів до категорій.

Таблиця 3.18 – Таблиця БД recipe_to_category

Поле	Тип(Довжина)	Опис
id_recipe	int(12)	id рецепта
id_category	int(12)	id категорії

Таблиця 3.18 містить в собі інформацію про належність усіх рецептів до своїх категорій. Один рецепт може належати до кількох категорій.

Таблиця БД steps_to_recipe, показана в таблиці 3.19, що містить інформацію про покроковому приготуванні рецептів.

Таблиця 3.19 – Таблиця БД steps_to_recipe

Поле	Тип(Довжина)	Опис
id (Primary Key index) AUTO_INCREMENT	int(12)	id кроку
id_recipe	int(12)	id рецепту до якого належить крок
image	varchar(254)	ілюстрація якщо є
text_ru	text	Опис кроку (російська мова)
text_uk	text	Опис кроку (українська мова)
text_en	text	Опис кроку (англійська мова)
time	int(11)	Час приготування якщо є

Таблиця 3.19 містить інформацію про всі етапи приготування страви в рецептах.

Таблиця БД support, показана в таблиці 3.20, яка містить дані з проханнями користувачів і відповідями на них.

Таблиця 3.20 – Таблиця БД support

Поле	Тип(Довжина)	Опис
s_id (Primary Key) AUTO_INCREMENT	int(8)	id запису
account_id	int(8)	id запис користувача
app_inh	enum(list)	Додаток до якого відноситься запис
email	varchar(65)	Адреса користувача
lang	enum('en', 'ru', 'uk')	Мова додатку
name	varchar(65)	Ім'я користувача
category	tinyint(3)	Категорія звернення
recall	text	Дата відповіді
answer	text	Текст відповіді
time CURRENT_TIMESTAMP	timestamp	Час і дата відправлення звернення
status	tinyint(2)	Тип звернення

Таблиця 3.20 містить інформацію про запити користувачів і стан їх обробки.

Таблиця БД tags, показана в таблиці 3.21 яка містить назву тегів.

Таблиця 3.21 – Таблиця БД tags

Поле	Тип(Довжина)	Опис
tag_id (Primary Key) AUTO_INCREMENT	int(10)	id тегу
text_ru	varchar(256)	Найменування тегу (російська мова)
text_uk	varchar(256)	Найменування тегу (українська мова)
text_en	varchar(256)	Найменування тегу (англійська мова)

Таблиця 3.21 містить в собі всі теги.

Таблиця БД videos показана в таблиці 3.22, яка містить інформацію в усіх відео доступних в системі.

Таблиця 3.22 – Таблиця БД videos

Поле	Тип(Довжина)	Опис
v_id (Primary Key) AUTO_INCREMENT	int(12)	id відео
code	varchar(45)	id відео на YouTube
name_ru (index)	text	Найменування відео (російська мова)
name_uk (index)	text	Найменування відео (українська мова)
name_en (index)	text	Найменування відео (англійська мова)

Продовження таблиці 3.22

Поле	Тип(Довжина)	Опис
text_ru (index)	text	Текст відео (російська мова), якщо є
text_uk (index)	text	Текст відео (українська мова), якщо є
text_en (index)	text	Текст відео (англійська мова), якщо є
id_article	int(12)	id новини або статті якщо є
id_recipe	int(12)	id рецепта якщо є

Таблиця 3.22 містить в собі інформацію про всіх відео і їх приналежність до статей/новин та рецептів.

3.5 Висновки до розділу

В результаті аналізу різних баз даних для реалізації системи була обрана база даних MySQL тому, що вона максимально відповідає таким вимогам як безпеку, швидкодія і відкритість коду.

4 ВИБІР МОВ ПРОГРАМУВАННЯ

4.1 Вимоги для проектування веб системи

Необхідно, щоб до системи можна було б доступитися з браузера на мобільному, планшетному і десктопном пристрої, за допомогою API для мобільних додатків.

Система не повинна повністю перезавантажуватися для отримання інформації, оновленню підлягає тільки та його частина, з якої йде робота

Необхідно, щоб система сама підлаштовувалась під розширення браузера і, в разі зміни розширення браузера, змінювалась без перезавантаження системи.

тавикористовуються.

Система повинна давати можливість працювати одночасно не менше 50 користувачам.

4.2 Аналіз мов і методів для створення системи

Для розробки веб-системи були переглянуті всі мови програмування, які відповідають вимогам розробки веб-системи.

4.2.1 Дослідження мов програмування веб-системи

Для створення веб-системи у розробників є великий арсенал мов програмування:

- JSP
- PHP;
- ASP;
- ASP.Net;
- Perl.

Сьогодні найчастіше використовуються (з великим відривом від інших) мови PHP і ASP.Net [7].

Переваги PHP:

- легкість вивчення;
- PHP - це мова, яка дає можливість динамічно виводити HTML-розмітку і іншу інформацію (наприклад, заголовки для Cookie та ін.);
- швидкість. Використовуючи модулі PHP, ми знаходимося в загальному просторі пам'яті PHP;
- робота з сесією. Існує два стандартний механізми роботи з сесією, а саме мемкеш і файли, де будь-який реквест це новий короткоживучих процес;
- вставка PHP-коду в звичайний статичну HTML-розмітку;

Недоліки:

- імовірність отримання низькоякісного продукту;
- не існує многопоточності і асинхронності.

ASP.Net. Переваги:

- менше шанс отримати низькоякісний продукт;
- многопоточність та асинхронність;

Недоліки:

- ASP повільніше працює, ніж PHP. При необхідності виведення інформації за запитом задіє об'єкт Response, А для звернення до бази даних викличе ще один об'єкт, для роботи з Вайль ще один. Таке накопичення об'єктів уповільнює роботу ASP додатки [8];
- при роботі з сесією в режимі In-proc дані сесії будуть зберігатися в пам'яті процесу, в домені додатку. Тому додаток працює довше [9];
- кожна сторінка складається з двох файлів: файлу з HTML-розміткою і ASP.Net-контроль.

Проаналізувавши всі переваги і недоліки цих двох мов, для розробки системи був обраний мову PHP

4.2.2 CSS3 каскадна таблиця стилів

В основному цю таблицю використовують для опису, оформлення зовнішнього вигляду веб-сторінок, які написані за допомогою мов розмітки

Цілі і завдання CSS:

- відокремити код html від коду опису стилів і дизайну;
- зробити дружній інтерфейс;
- зробити сайт більш привабливим (в плані дизайну).

Для різних розширень екрану без перезавантаження сторінки дозволяє робити додаткові налаштування.

Переваги використання CSS:

- простий синтаксис мови;
- загальний спосіб зміни дизайну на сайті;
- легкість зміни дизайну. Для цього достатньо змінити стиль в одному місці і він зміниться на всіх сторінках сайту.

4.2.3 JavaScript мова програмування

JavaScript використовується в клієнтській частини веб-додатків: клієнт-серверних програм, де клієнтом є браузер, а сервером - веб-сервер, які мають розподілену між сервером і клієнтом логіку. Обмін інформацією в веб-додатках виконується по мережі. Одним з достоїнств такого підходу є те, що клієнти не залежать від певної операційної системи користувача, і тому веб-додатки є кроссплатформенними сервісами.

Також перевагою JavaScript є те, що сценарії здійснюється на машині користувача, що зменшує навантаження сервера.

JavaScript використовується в AJAX, поширеному підході до створення користувацьких інтерфейсів веб-додатків, які перебувають в «фоновому» асинхронному обміні даними браузера з веб-сервером. У підсумку, веб-сторінка не перезавантажується повністю при оновленні даних і інтерфейс веб-додатки

стає швидше, ніж це відбувається при звичайному підході (без застосування AJAX) [10].

4.2.4 Мова AJAX технологія звернення до сервера без перезавантаження сторінки

AJAX - концепція використання декількох суміжних технологій, а не самостійна технологія. AJAX має два основних принципи:

- застосування DHTML для динамічної зміни змісту сторінки;
- застосування технології динамічного звернення до сервера «на льоту», без перезавантаження всієї сторінки повністю, наприклад з використанням XMLHttpRequest (основний об'єкт).

При використанні AJAX значно скорочується трафік роботи з веб-додатком внаслідок чого, досить завантажити тільки змінену частину або взагалі тільки отримати/передати набір даних в форматі JSON або XML, замість завантаження всієї сторінки, а потім замінити вміст сторінки за допомогою JavaScript [11].

4.2.5 Текстовий формат обміну даними JSON

Завдяки своїй стислості в порівнянні з XML, формат JSON може бути найбільш відповідним для серіалізації складних структур. Говорячи про веб-додатках, в цьому сенсі, він переважний в задачах обміну даними, як між браузером і сервером (AJAX), так і між самими серверами (програмні HTTP-сполучення).

Використовуючи для відправки та отримання даних між сервером і додатком він дуже зручний.

JSON організований на двох структурах даних:

- впорядкований список значень. У більшій частині мов це здійснене як вектор, масив, список або послідовність [12];

- колекція пар ключ/значення. У різних мовах, ця концепція представляється як запис, об'єкт, словник, структура, хеш, іменований список або асоціативний масив.

4.2.6 jQuery бібліотека JavaScript

Аналогічно тому, як CSS відокремлює візуалізацію від структури HTML, JQuery відокремлює поведінку від структури HTML. Наприклад, замість прямої вказівки на обробника події натискання кнопки управління передається JQuery, яка розпізнає кнопки і потім перетворює його в обробник події кліка. Такий поділ дії і структури також називається принципом ненав'язливого JavaScript.

На сучасному етапі користуються всюди, де використовують JavaScript.

Роботу з jQuery можна розділити на 2 типи:

- виклик глобальних методів у об'єкта \$, наприклад, зручних ітераторів по масиву;
- отримання jQuery-об'єкта за допомогою функції \$ (). Наприклад, можна отримати jQuery-об'єкт всіх елементів HTML, які потрапляють під критерій, передавши в неї CSS-селектор, і далі діяти з ними, використовуючи різні методи jQuery-об'єкта. У разі, якщо метод не повинен повертати ніякого значення, він повертає посилання на jQuery об'єкт, який дає можливість вести ланцюжок викликів методів згідно з концепцією текучого інтерфейсу.

4.3 Опробування системи

4.3.1 Опробування функцій системи

Всі функції системи були ретельно перевірені:

- перевірка всіх форм відправки;
- тестування пошукової роботи;
- випробувано завантаження файлів на сервер

- перевірено робота всіх сторінок і посилань [13].

Випробування пройшли успішно

4.3.2 Перевірка верстки на різних пристроях

При тестуванні ресурсу використовували різні дозволи екрану за рахунок Chrome Developers Tool.

Спочатку змінили розширення на розширення відповідне планшетному пристрою iPad 768x1024. Підсумок відображений на рисунку 4.1.

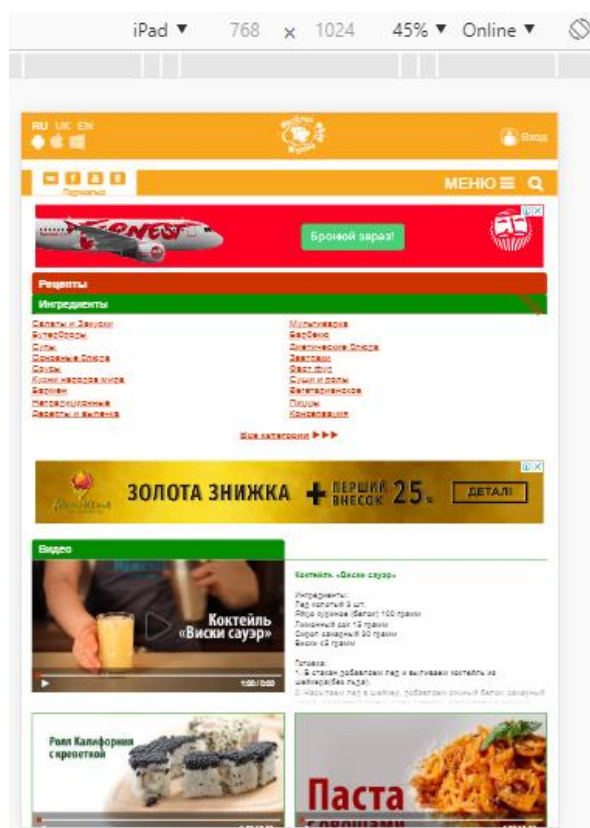


Рисунок 4.1 – Розширення екрану 768x1024

Система відразу перебудувалася, потім була проведена імітація повороту пристрою на 180 градусів. Підсумок видно на рисунку 4.2.

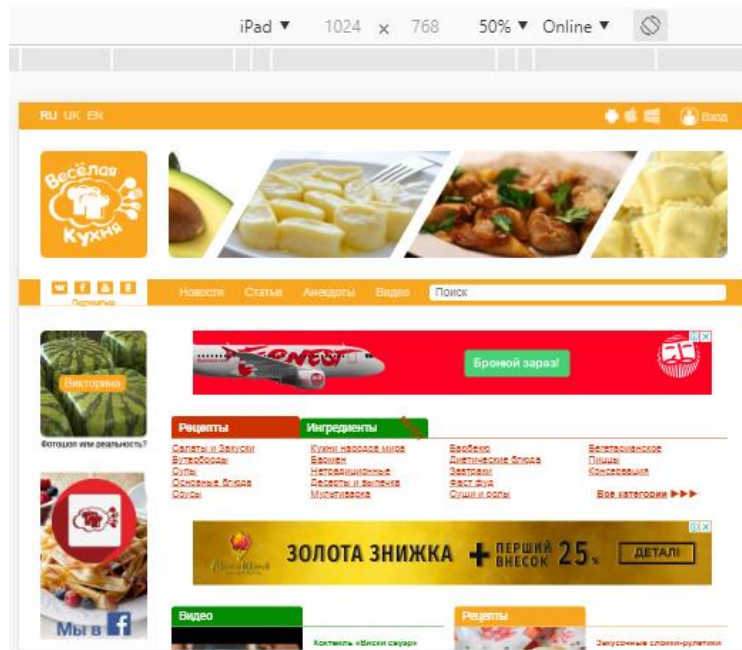


Рисунок 4.2 – Розширення екрану 1024x768

Аналогічні випробування були зроблені при імітації мобільного пристрою Samsung Galaxy S5. Він має дозвіл екрана 360x640 і при повороті на 180 градусів - 640x360. Адаптація системи під всі розширення проходить успішно.

4.3.3 Опробування безпеки

Система була перевірена на чутливість до XSS і sql-injection.

XSS дійсно має високий ступінь небезпеки, оскільки її можуть використовувати для зміни DOM-моделі сайту, що в свою чергу дасть можливість викрасти облікові дані адміністратора сайту і отримати повний контроль над вразливим додатком [14].

Система була протестована на наявність XSS і sql-injection уразливості. Уразливості не знайдено.

При проектуванні системи були передбачені завдання з пошуку спеціальних символів у всіх рядках в:

- url параметрам;

- будь-яких вводяться користувачем;
- cookie [15].

Був доданий SSL сертифікат для доповнення безпеки системи.

HTTPS - захищена версія протоколу HTTP. Він починає діяти після установки SSL-сертифіката і зашифровує особисту інформацію, перед передачею її власнику сайту.

Принцип роботи SSL-шифрування. У основі будь-якого методу шифрування знаходиться ключ. Ключ - це спосіб зашифрувати або розшифрувати що-небудь, наприклад, повідомлення. В роботі SSL-сертифіката беруть участь три ключа: приватний, публічний і сеансовий [16].

В результаті тестування системи було встановлено, що система повністю захищена і безпечна.

4.3.4 Тестування ефективності системи

Серед вимог для розробки веб системи, які описані в пункті 4.1, була можливість одночасної роботи з системою не менше 50 користувачам.

Для тестування навантаження обраний ресурс loadly.com. На рисунку 4.3. показані результати тестування

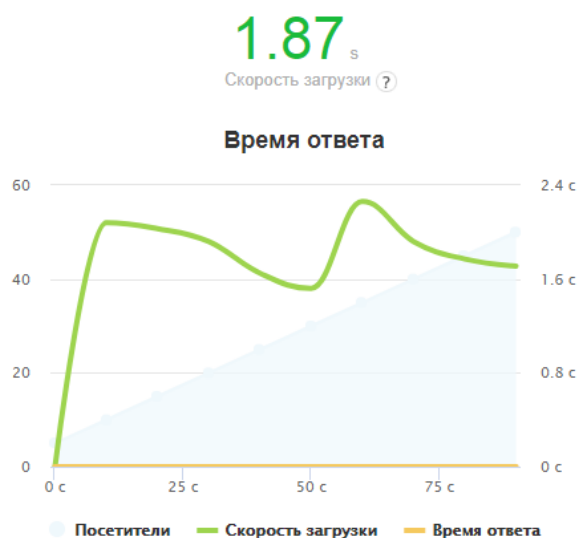


Рисунок 4.3 – Результати тестування навантаження.

Тестування показало, що при використанні системи 50 користувачами одночасно швидкість завантаження системи в середньому становить 1.87 секунди, що відповідає вимогам поставленого завдання.

4.4 Висновок до розділу

Для коректної розробки інформаційно-довідкової системи в кулінарній галузі необхідно було спочатку встановити чіткі вимоги до цієї системи. Після цього провести аналіз мов і методів програмування інформаційно-довідкової системи. Розглянути всі достоїнства і недоліки цих інструментів програмування. Після проведеного аналізу були обрані такі мови і бібліотеки програмування для реалізації системи::

- PHP;
- CSS;
- JavaScript;
- AJAX;
- JSON;
- JQuery.

Після написання інформаційно-довідкової системи в кулінарній галузі, отриману систему протестували на виконання всіх вимог. Система була перевірена на функціональність системи, верстку системи на різних пристроях, перевірена безпека системи, а також її ефективність. Тестування показало що система відповідає всім вимогам поставленого завдання.

5 СТВОРЕННЯ СИСТЕМИ

Спочатку роботи з системою була створена база даних.

На рисунку 5.1. зображена створена база даних.

Table	Action	Rows	Type	Collation	Size	Overhead
admin_account	★ Browse Structure Search Insert Empty Drop	7	InnoDB	utf8_general_ci	16 KiB	-
anekdot	★ Browse Structure Search Insert Empty Drop	471	MyISAM	utf8_general_ci	224.5 KiB	-
articles	★ Browse Structure Search Insert Empty Drop	1,144	MyISAM	utf8_general_ci	11 MiB	-
articles_to_tag	★ Browse Structure Search Insert Empty Drop	2,789	MyISAM	utf8_general_ci	25.5 KiB	-
bookmarks	★ Browse Structure Search Insert Empty Drop	250	MyISAM	utf8_general_ci	13.8 KiB	-
category_ingredients	★ Browse Structure Search Insert Empty Drop	25	MyISAM	utf8_general_ci	3.7 KiB	-
category_recipe	★ Browse Structure Search Insert Empty Drop	157	MyISAM	utf8_general_ci	16.7 KiB	-
favorites	★ Browse Structure Search Insert Empty Drop	149,555	InnoDB	utf8_general_ci	16 KiB	-
food_diary	★ Browse Structure Search Insert Empty Drop	2,849	InnoDB	utf8_general_ci	16 KiB	-
food_history	★ Browse Structure Search Insert Empty Drop	11,427	InnoDB	utf8_general_ci	16 KiB	-
food_info	★ Browse Structure Search Insert Empty Drop	161	InnoDB	utf8_general_ci	16 KiB	-
ingredients	★ Browse Structure Search Insert Empty Drop	3,012	MyISAM	utf8_general_ci	1.6 MiB	-
ingredients_to_recipe	★ Browse Structure Search Insert Empty Drop	63,616	MyISAM	utf8_general_ci	2.1 MiB	-
log	★ Browse Structure Search Insert Empty Drop	17,754	InnoDB	utf8_general_ci	16 KiB	-
recipe	★ Browse Structure Search Insert Empty Drop	7,637	MyISAM	utf8_general_ci	1.5 MiB	-
recipe_like	★ Browse Structure Search Insert Empty Drop	27,213	MyISAM	utf8_general_ci	718.8 KiB	-
recipe_to_category	★ Browse Structure Search Insert Empty Drop	13,435	MyISAM	utf8_general_ci	119.1 KiB	-
steps_to_recipe	★ Browse Structure Search Insert Empty Drop	39,103	MyISAM	utf8_general_ci	24.6 MiB	-
support	★ Browse Structure Search Insert Empty Drop	114	MyISAM	utf8_general_ci	16.1 KiB	-
tags	★ Browse Structure Search Insert Empty Drop	30	MyISAM	utf8_general_ci	3.3 KiB	-
videos	★ Browse Structure Search Insert Empty Drop	75	MyISAM	utf8_general_ci	26.4 KiB	-
21 tables	Sum	340,824	MyISAM	utf8_general_ci	41.9 MiB	0 B

Рисунок 5.1 – створена база даних

Після цього йде підключення до бази даних в файлі bdConnection.php, яке в свою чергу бере дані з siteConfig.php (додаток А).

Все управління всередині системи відображено в файлі init.php (додаток Б), який відповідно використовує модулі всієї системи, такі як functions.php (Додаток В) и site_functions.php (Додаток Г).

Після того як підключили базу даних і всі файли настройки, файл index.php головна сторінка системи почала працювати, що видно на малюнку 5.2

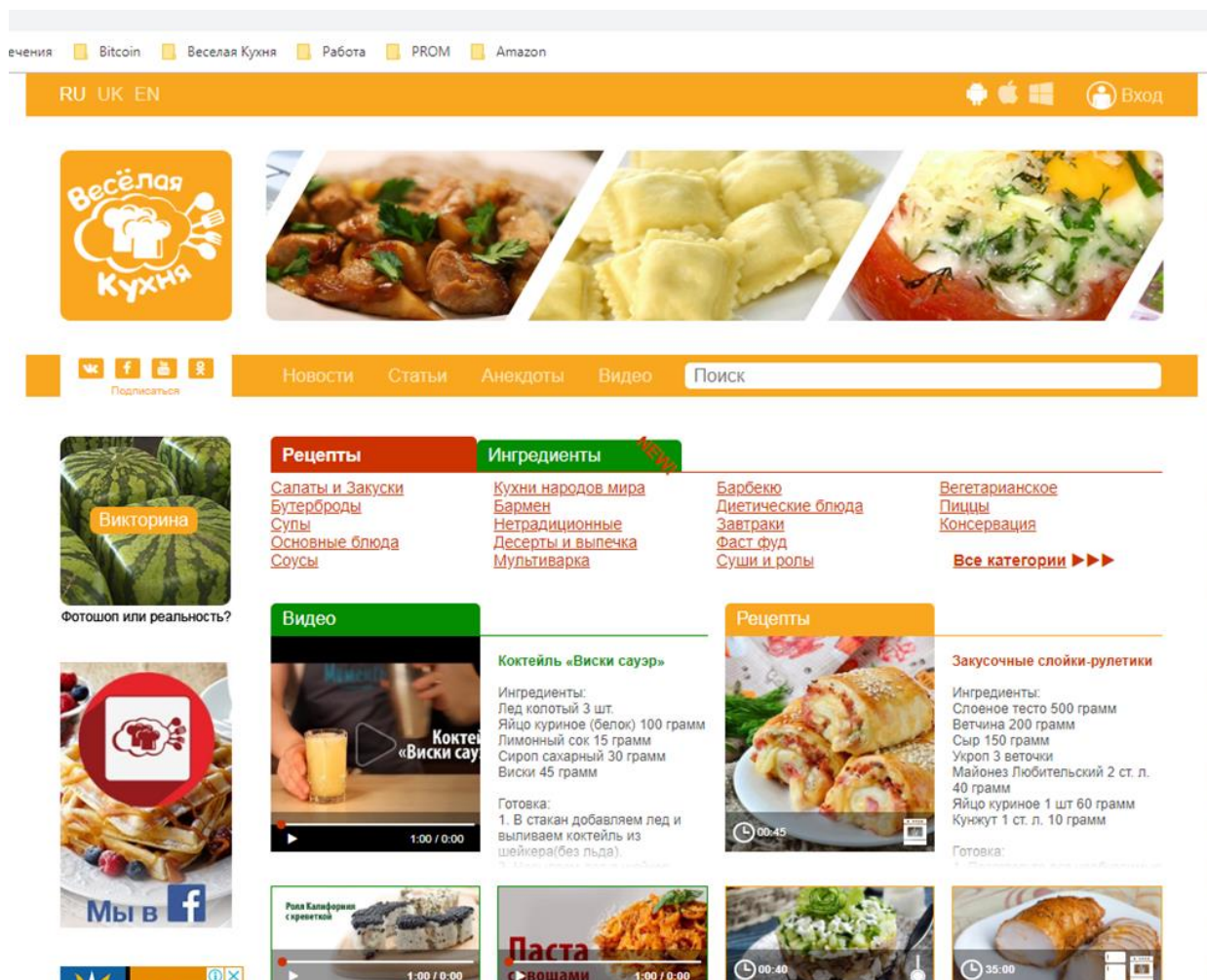


Рисунок 5.2 – Головна сторінка системи

Потім у файлі .htaccess (Додаток Д) налаштували помилки 403, 404, і 50X, опис яких знаходиться в 403.php (додаток Д), 404.php (Додаток Ж) та 50X.php (Додаток Ж).

403 Forbidden — Сервер розпізнав запит, але він не виконує його через обмеження в доступі для клієнта до зазначеного ресурсу. Іншими словами, клієнт не уповноважений здійснювати операції з замовленим ресурсом. [56]

На малюнку 5.3 відображена 403 помилка.



Рисунок 5.3 – вид системи з 403 помилкою

Помилка 404 або Not Found («не знайдено») - тандартний код відповіді HTTP про те, що клієнт був здатний спілкуватися з сервером, але сервер не може знайти дані, згідно із запитом. Помилку 404 треба відрізняти від помилки «Сервер не знайдений» тобто, помилку, яка вказує на обмеження доступу до сервера. Помилка 404 говорить про те, що запитуваний ресурс може бути доступний в майбутньому, тільки не гарантує наявності попереднього змісту. [56]

На рисунку 5.4 показано вид системи з 404 помилкою.

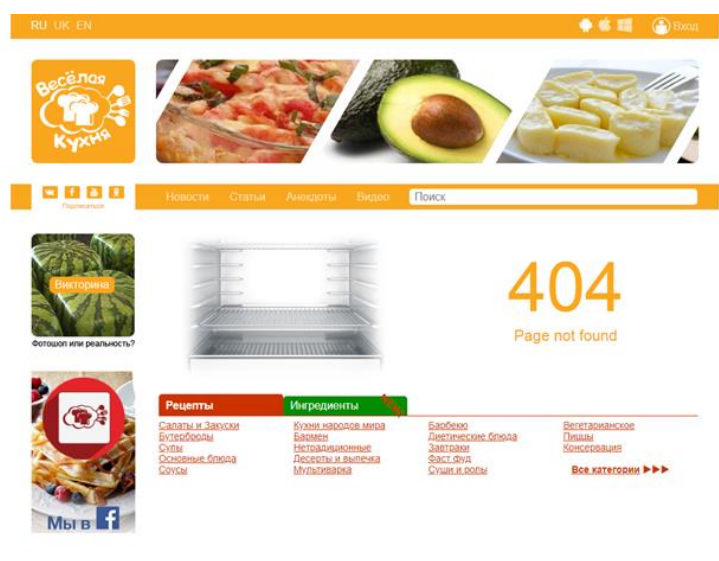


Рисунок 5.4 –вид системи з 404 помилкою

Коди 50X відведені для випадків невдалого виконання операції з вини сервера. У всіх випадках, крім використання методу HEAD, сервер зобов'язаний включати в тіло повідомлення пояснення, яке клієнт відобразить користувачеві. [56]

На малюнку 5.5 відображена 50X помилка.

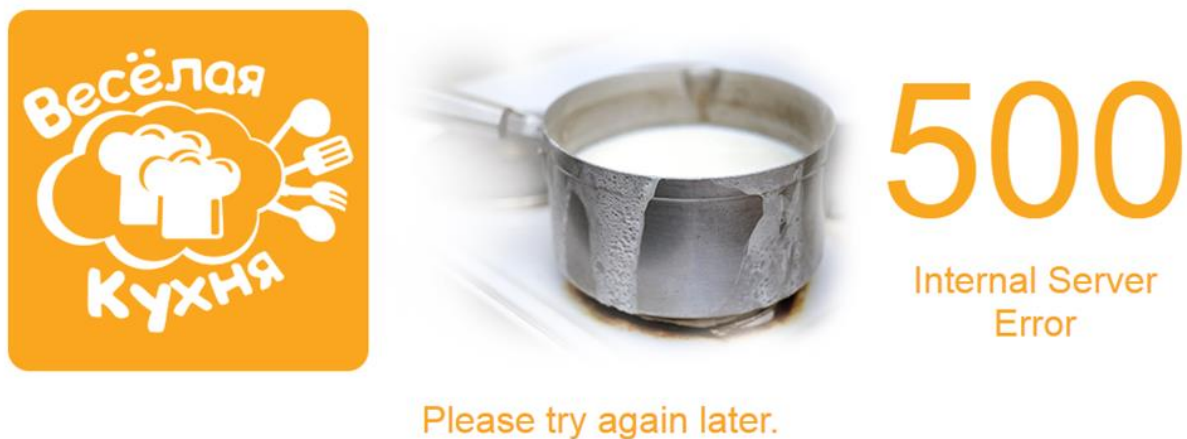


Рисунок 5.5 –вид системи з 50X помилкою

У додатку К наведено відображення парсинга JSON коду, а в додатку Л - лістинг виведення цього коду.

5.1 Висновок до розділу

В результаті розробки даного розділу була створена повністю робоча інформативно-довідкова система в кулінарній галузі.

6 ІНСТРУКЦІЯ КОРИСТУВАЧА

На рисунку 6.1 показана головна сторінка сайту, яка дозволяє ознайомитися з його розділами.

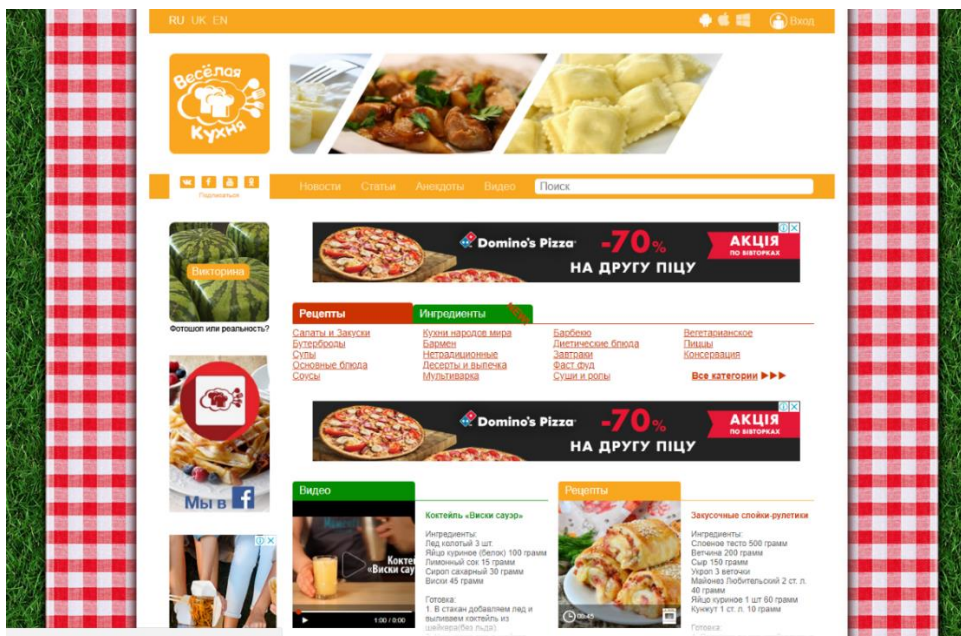


Рисунок 6.1 – Головна сторінка сайту

З головної сторінки можна авторизуватися на сайті, вибрати мову інтерфейсу та потрапити в будь-який розділ.

На рисунку 6.2 показана форма авторизації на сайті.

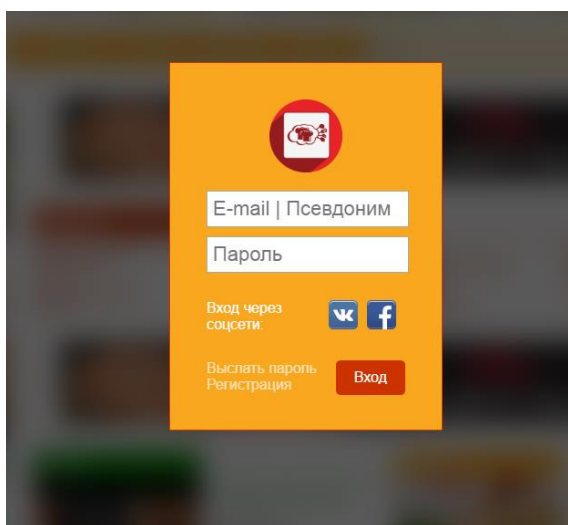


Рисунок 6.2 – Форма авторизації

На рисунку 6.3 показан пошук по системі.

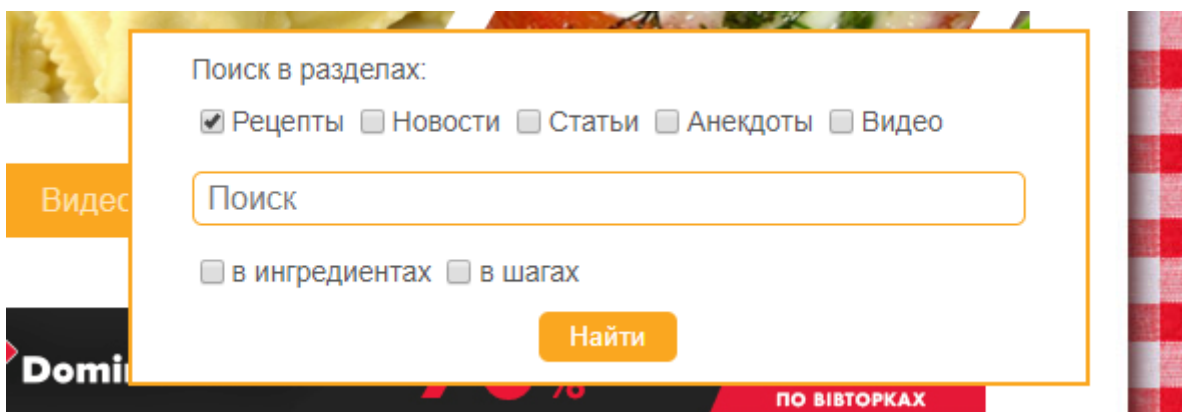


Рисунок 6.3 – Пошук по системі


На рисунку 6.4 показана інформація про рецепт.

Рецепты **Все категории** ▶ Десерты и выпечка ▶ Пирожное и десерты

Панна-Котта 🖨

⌚ 9 ч. 40 мин. 👤 8 чел
 На 100гр - 170 ккал: белки - 2.86, жиры - 12.54, углеводы - 11.36

📱 📺 📺 📺 📺 📺



Блюдо

Нектар из маракуйи	100 грамм
Усики горошка	5 грамм
Лепестки роз	10 грамм
Апельсин красный	1/2 шт. 100 грамм
Молоко	500 грамм
Сливки 33% (классические)	185 грамм
Сметана 40% (жирная)	75 грамм
Миндаль сладкий	20 грамм
Яблоки сушеные	20 грамм
Сыр Филадельфия	150 грамм
Ваниль	1 стручок
Желатин листовой	8 шт.
Мята свежая	10 грамм
Сахар-Песок	65 грамм
Шафран	1 грамм
Крем бальзамический	10 грамм

1/8

В сотейник влить молоко и добавить шафран — совсем чуть-чуть, исключительно для цвета. Поставить на огонь, довести температуру молока до 90 градусов и держать так два-три часа, чтобы шафран настоялся в мягкий кремово-оранжевый цвет.

2/8

Рисунок 6.4 – Інформація про рецепт

6.1 Висновок до розділу

При конструюванні інтерфейсу системи керувалися правилом, що він повинен бути таким зрозумілим, щоб не було потрібно пояснень. Для того, щоб система запам'ятала відвідувача необхідно зареєструватися в системі.

Авторизований користувач має право шукати, переглядати інформацію, а також додавати цю інформацію до обраного, а неавторизований користувач має права тільки на пошук і перегляд інформації системи. Адміністратор має право додавати нову інформацію.

7 РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

Стартап або стартап-компанія (від англ. Start-up - запускати) - це вид бізнесу, який спрямований на отримання доходу шляхом здійснення принципово нової ідеї.

На відміну від звичайного бізнесу, який людина тільки відкриває, стартап має інноваційну основу, тобто відкривається бізнес, якого раніше не було взагалі, або іншими словами - це бізнес, якого раніше не існувало. Ідея стартапа полягає в чомусь новому: новому якомусь продукті, новій технології, в тому, чого раніше не існувало.

Термін стартап застосуємо абсолютно для всіх галузей економіки. У сферах ІТ технологій, інформаційних технологій і інтернет-проектів, наприклад, стартапами називаються нові компанії, які тільки з'являються або планують створити.

Стандартна структура стартап проекту зображена на рисунку 7.1.

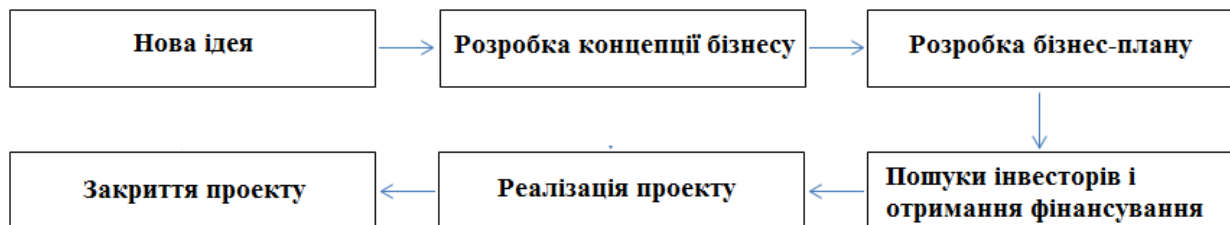


Рисунок 7.1 – Стандартна структура стартап проекту

ІТ-стартапи ніяк не прив'язані до різних маркетингових і піар-методів, які ставлять собі за мету швидку розкрутку фірми. ІТ-стартапи мають далекосяжні плани і створюються на тривалий термін. Наприклад, створення продукту протягом одного-двох років, і тільки після цього вихід на ринок та просування в інтернеті і не тільки. [31]

7.1 Історія створення та розвитку стартапів

Слово стартап останні десятиліття стало означати прорив, який тісно пов'язаний з новими технологіями. Раніше стартап мав інший сенс і називався - гаражний бізнес. Це сталося в далекому 1939 році. Тоді два студента Хьюлетт і Паккард так назвали свій проект. Start up означає починати, запускати. Проект виявився досить успішним: продукцією HP користуємося ми і сьогодні.

Терміном стартап стали активно користуватися за часів бульбашки доткомів, коли створювалася велика кількість інтернет-компаній, які займалися новітніми технологіями і наданням послуг, яких ніколи до цього не існувало. Нові проекти, створювані в галузях високих технологій, часто називають хайтек-стартап.

Для повного розуміння, що таке стартапи давайте звернемося до джерел. Термін «стартап» вперше з'явився в США в 1939 році. Саме тут, біля міста Сан-Франциско, в долині Санта-Кларі (Каліфорнія), були сконцентровані майже всі підприємства і фірми, які займалися розробками в сфері високих технологій. Тоді ж студенти Стенфордського університету Девід Паккард і Вільям Хьюлетт, створювали тут свій, як їм здавалося, невеликий проект, і назвали вони його стартапом. Згодом цей стартап перетворився в таку величезну і дуже успішну компанію Хьюлетт-Паккард.

Для повного розуміння, що таке стартапи давайте звернемося до джерел. Термін «стартап» вперше з'явився в США в 1939 році. Саме тут, біля міста Сан-Франциско, в долині Санта-Кларі (Каліфорнія), були сконцентровані майже всі підприємства і фірми, які займалися розробками в сфері високих технологій. Тоді ж студенти Стенфордського університету Девід Паккард і Вільям Хьюлетт, створювали тут свій, як їм здавалося, невеликий проект, і назвали вони його стартапом. Згодом цей стартап перетворився в таку величезну і дуже успішну компанію Хьюлетт-Паккард.

Лише тільки інвестори зрозуміли які перспективи дають ці нові ідеї, і що в них можна вкласти трохи грошей і стати мільйонерами, стартапи стали набирати популярності.

7.2 Загальна інформація про стартап

Життєвий цикл стартапу. На початку роботи над будь-яким стартапом необхідно створити його прототип.

Потім прототип стає повноцінним продуктом, змінюючись і розвиваючись. Протягом усього цього часу стартап залучає інвестиції, команда стартапу зростає, а складність створюваного продукту збільшується.

Кінцевою метою створення стартапу є або продаж його якійсь великій корпорації або виведення його акцій на біржу і тоді продовження роботи в якості окремої компанії.

Хто ж робить стартапи? Так як хороший стартап є бізнесом, то "стартапер" - той хто пише його, є, в деякому сенсі підприємець.

Стартапер може бути не обов'язково бізнесменом, він може займатися дизайном або розробкою, і може бути економістом чи інженером.

Часто один стартап роблять двоє людей: підприємець і технар. Але буває, що стартап робить і одна людина, яка займається як розробкою, так і просуванням свого продукту.[32]

7.3 Інформаційна карта проекту

У таблиці 7.1 показана інформаційна карта стартап-проекту. Це типовий спосіб знайомства з ідеєю проекту для замовників, спонсорів і всіх інших зацікавлених сторін. Це дуже важливо враховувати при заповненні, тому що формує перше враження про розроблюваний продукт.

Таблиця 7.1 – Інформаційна карта проекту

Назва проекту	Інформаційно-довідкова система в кулінарній галузі
Анотація	Завдяки дослідженню інформаційно-довідкових систем вдалося реалізувати інформаційно-довідкову систему в кулінарній галузі. Для доступу до інформаційно-довідкової системи в кулінарній галузі знадобилося вибрати та реалізувати веб-додаток і додатки для мобільних та планшетних пристроїв. Для реалізації інформаційно-довідкової системи в кулінарній галузі були проаналізовані і обрані мови програмування, які оптимально підходять для здійснення цього завдання.
Термін реалізації	6-7 місяців
Необхідні ресурси	<p>Фінансові :</p> <ul style="list-style-type: none"> - заробітна плата працівникам; - оренда приміщення; - маркетингові дослідження. <p>Матеріальні:</p> <ul style="list-style-type: none"> - персональні комп'ютери та їх комплектуючі 5 шт. <p>Людські:</p> <ul style="list-style-type: none"> - програміст веб ресурсі; - програміст мобільних та планшетних додатків; - дизайнер; - менеджер/концепщик; - копірайтер.
Опис проблеми, яку вирішує проект	<p>Сьогодні інформаційні технології роблять роботу в сфері кулінарії якісніше і доступніше для великої кількості людей, що цікавляться.</p> <p>Довідкові системи дозволяють впоратися з будь-якими завданнями для задоволення запитів споживачів в отриманні необхідної інформації.</p> <p>Пошук необхідної інформації в інформаційно довідкових системах відбувається значно швидше, ніж в паперовому аналогу.</p>
Головні цілі та завдання проекту	Виявлення актуальних проблем, які виникають при створенні автоматизованих інформаційно-довідкових систем, побудова поліпшеної моделі інформаційно-довідкової системи рецептів кулінарних блюд для підвищення ефективності процесу отримання інформації для користувачів, а також розробка зовнішніх додатків до неї.

Продовження таблиці 7.1

Назва проекту	Інформаційно-довідкова система в кулінарній галузі
Очікувані результати	Розробка конкретних аспектів інформаційно-довідкової системи, підвищення ефективності та розширення можливостей використання в сфері роботи кулінарної галузі як за допомогою десктопних пристроїв, так і з мобільних і планшетних пристроїв.

7.4 Команда стартап-проекту

В таблиці 7.2 перелічені функціональні обов'язки виконавців проекту

Таблиця 7.2 – Посадові обов'язки виконавців проекту

Посада	Функціональні обов'язки
Менеджер/концепщик проекту	<ul style="list-style-type: none"> - постановка задач - обговорення термінів - контроль за встановленими строками - контроль за виконанням поставлених завдань - тестування системи - створення маркетингового плану
Програміст веб ресурсу	Створення: <ul style="list-style-type: none"> - бази даних; - веб ресурсу; - API для взаємодії з мобільними та планшетними додатками.
Програміст мобільних та планшетних додатків	Створення: <ul style="list-style-type: none"> - додатку на iOS для мобільних та планшетних пристроїв; - додатку на Android для мобільних та планшетних пристроїв.
Дизайнер	<ul style="list-style-type: none"> - створення UX/UI дизайну веб ресурсу; - створення UX/UI дизайну для додатку мобільних та планшетних пристроїв.
Копирайтер	Наповнення бази системи: <ul style="list-style-type: none"> - додавання/копіювання та переробка рецептів; - додавання/копіювання та переробка інгредієнтів; - додавання/копіювання та переробка новин; -

Продовження таблиці 7.2

Посада	Функціональні обов'язки
Копирайтер	- додавання/копіювання та переробка статей.

7.5 Маркетинговий аналіз стартап-проекту

7.5.1 Маркетингова концепція товару

З огляду на те, що товар в маркетингу - це поєднання матеріальних і нематеріальних характеристик, які представляються для задоволення потреб і забезпечують очікувані вигоди як для споживачів, так і для товаровиробників. Товаровиробник одержує вигоду у вигляді доходу і прибутку, або в придбанні інших ринкових цілей при продажу товару. Споживач купуючи товар - набуває набір властивостей, які задовольняють його потреби, або вирішують якісь його проблеми.

Найпростіше розуміння концепції товару - це видить в ньому тільки набір фізичних параметрів.

За Т. Левітом, є ще розширена концепція товару, в яку включені такі заходи, які зроблять товар ще ціннішим в очах споживачів, тобто, забезпечать підприємцю можливість залучення споживачів або допоможуть йому утримати споживача, як свого клієнта.[33]

7.5.2 Опис ідеї проекту

Створення Інформаційно-довідкової системи в кулінарній галузі яка буде якісніша і доступніша для великої кількості людей. Виявивши актуальні проблеми, що виникають при створенні автоматизованих інформаційно-довідкових систем, побудувати поліпшену модель інформаційно-довідкової системи рецептів кулінарних страв для підвищення ефективності процесу отримання інформації для користувачів, а також розробка зовнішніх додатків до неї.

У таблиці 7.3 знаходиться опис ідеї стартап-проекту.

Таблиця 7.3 – Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Створення Інформаційно-довідкової системи в кулінарній галузі.	1. Використання з стаціонарних комп'ютерів, ноутбуків та планшетів.	Зручний та user friendly інтерфейс. Повна інформація про рецепти, інгредієнти. Їх властивості та вміст білків, жирів, вуглеводів та калорій на 100 грамів. Велика кількість новин та статей з кулінарної галузі. Зручне зберігання, поширення чи друк. Доступ до калоризатору.
	2. Використання з мобільних та планшетних пристроїв.	Зручний та user friendly інтерфейс адаптований для мобільних та планшетних пристроїв. Повна інформація про рецепти, інгредієнти. Їх властивості та вміст білків, жирів, вуглеводів та калорій на 100 грамів. Велика кількість новин та статей з кулінарної галузі. Зручне зберігання чи поширення. Доступ до калоризатору.
	3. Калоризатор	Дає можливість вибрати режим питания(діти) під себе та дотримуватися його за вдяки величезній базі рецептів та інгредієнтів. Можливість використовувати як на стаціонарних комп'ютерах чи ноутбуках, так і на мобільних та планшетних пристроях.

У таблиці 7.4 знаходиться визначення сильних, слабких та нейтральних ідеї проекту.

Таблиця 7.4 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту.

№ з/п	Техніко-економічні характеристики ідеї	(потенційні) концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна)
		Мій проект	Система russianfood.com	Система povarenok.ru	Система gotovimdoma.ru			
1.	Інформативність рецептів	Гарна	Нейтральна	Нейтральна	Нейтральна			S
2.	Наявність інформації про інгредієнти	Є	Відсутня	Є	Відсутня			S
3.	Наявність інформації про кількість Б/Ж/В та ккал у страві	Є	Відсутня	Є	Відсутня			S
4.	Зручність інтерфейсу	Зручний	Зручний	Зручний	Незручний			S
5.	Наявність підтримки мобільних і планшетних пристроїв	Підтримує	Підтримує	Підтримує	Не підтримує			
6.	Наявність мобільного чи планшетного додатку	Є	Відсутній	Відсутній	Відсутній			S
7.	Присутність реклами	Достатньо	Забгато	Забгато	Забгато		N	

Продовження таблиці 7.4

№ з/п	Техніко-економічні характеристики ідеї	(потенційні) концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна)
		Мій проект	Система russianfood.com	Система povarenok.ru	Система gotovim-doma.ru			
8.	Наявність новин чи статей	Є	Є	Відсутній	Є		N	
9.	Наявність калоризатору	Є	Відсутній	Відсутній	Відсутній			S

7.5.3 Технологічний аудит ідеї проекту

У таблиці 7.5 знаходиться технологічна здійсненність ідеї проекту.

Таблиця 7.5 – Технологічна здійсненність ідеї проекту.

№ з/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
		Технологія 1 Реалізація front end за допомогою мов програмування: - HTML - JavaScript - jQuery - AJAX - CSS Реалізація back end за допомогою мови програмування PHP	Розроблено	Доступна
		Технологія 2 СУБД mySQL	Розроблено Базу Даних	Доступна
2.	Створення API системи	Технологія 3 Для реалізації API використовуємо зв'язку PHP та JSON	Розроблено	Доступна

Продовження таблиці 7.5

№ з/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
3.	Створення додатку для мобільних та планшетних пристроїв	Технологія 4 Для додатку на Android використовували мову програмування JAVA. Для додатку на iOS використовували мову програмування SWIFT	Розроблено	Доступна
Обрана технологія реалізації ідеї проекту: HTML, JavaScript, jQuery, AJAX, CSS, PHP, JSON, JAVA, SWIFT.				

7.5.4 Маркетинговий план системи

Маркетинговий план - це розписаний порядок маркетингових дій і комунікацій, метою якого є досягнення довгострокових цілей компанії, з розрахунками всіх витрат, ризиками і стратегією.

Іноді власники компаній не надають належного значення ефективності такого плану, думаючи що це марна трата грошей і часу.

Продукт продається, клієнти є. Все добре. Але це не так. Завтра на ринок зайде якийсь новий гігант і відведе клієнтів.

Переваги маркетингового плану:

- видно картина майбутнього;
- зрозуміло, як потрібно розподілити ресурси;
- покращиться бізнес;
- побачите проблеми;
- передбачте результати;
- ліквідуєте недоліки.

Недоліки маркетингового плану:

- додаткові витрати;
- застарілі дані;
- похибка результатів;
- немає гарантій.

Маркетинговий план включає в себе:

- установка спільної мети плану;
- вибір відповідальних осіб за складання плану;
- безперервне оновлення (ведення) плану;
- формулювання цілей і установка термінів планування;
- деталізація дії для досягнення цілей;
- складання детального фінансового плану по кожній статті витрат;
- опис всіх дій в разі позапланових ситуацій.[34]

7.5.5 Впізнаваність системи

Словосполучення «впізнаваність бренду» означає вміння потенційних покупців упізнати відому торгову марку при виборі або купівлі товару. Це словосполучення включає в себе здатність споживачів швидко і легко розпізнавати бренд за окремими характеристиками.

Компетентність споживачів про торгову марку впливає на обсяги продажів товару, тому маркетингові кампанії повинні приділяти значну увагу підвищенню впізнаваності бренду. Від популярності торгової марки, її популярності залежить здатність продукції конкурувати на ринках товарів і послуг, а також перспективи довгострокового зростання бренду.

Різновиди оцінок рівня впізнаваності:

- 1) знання з підказками - покупець згадує бренд тільки при прямому контакті -бачить упаковку, логотип або чує назву або слоган;
- 2) знання без підказок - бренд досить добре запам'ятався споживачеві і викликає конкретний пов'язаний з ним ряд;

3) top of mind (вершина розуму) - це перші 3-4 бренди, які покупець згадає в певній товарній групі.[35]

Типи підвищення впізнаваності бренду в інтернеті:

- зовнішній контент-маркетинг;
- внутрішній контент-маркетинг;
- медійна реклама;
- співпраця з лідерами думок.

Переваги медійної реклами:

- digital генерує конверсії, швидше ніж це робить ТВ;
- ви контролюєте рекламну кампанію і в будь-який момент маєте право вносити зміни або зупинити показ;

- є можливість зробити більш точним таргетинг;
- краще розрахувати результати, наприклад, за охопленням;
- покращує стан сайту в пошуковій видачі (КМС);
- немає таких жорстких часових рамок, які є в ТВ-рекламі;
- максимально автоматизує хід просування бренду.

Переваги зовнішнього контент-маркетингу:

- дешевший інструмент в порівнянні з ТВ і медійною рекламою;
- підвищує SEO-показники;
- звільняє ресурси інхауси-команди;
- дозволяє збільшити число представників цільової аудиторії і підвищити вплив на них;

- часто не сприймається користувачем як пряма реклама і тому викликає більше довіри у нього.[36]

Шляхи розвитку внутрішнього контент-маркетингу:

- відеоблог;
- актуальний блог (чек-листи, кейси, статті, інфографіка);
- подкасти;
- офіційні сторінки бренду в соціальних мережах;
- Telegram-канал.

Цінність внутрішнього контент-маркетингу:

- посередників між вами і користувачем немає, взаємодія відбувається на ваших майданчиках;
- кількість якісних зворотних посилань збільшується;
- один з найбільш недорогих інструментів;
- ви контролюєте повністю процес взаємодії з користувачем і в будь-який момент можете внести зміни;
- можете взаємодіяти з аудиторією.

Цінність співпраці з лідерами думок:

- зростає довіра аудиторії;
- користувачі часто сприймають його як нативну рекламу;
- збільшується охоплення аудиторії;
- один з найбільш оперативних інструментів для спілкування з мілленіалами і поколінням Z;
- зростає довіра аудиторії;
- забезпечує оперативний і інформативний зворотний зв'язок з передплатниками.

Будь-яка рекламна кампанія - сукупність заходів, ефективність яких передається конкретними параметрами. Сюди ж відносяться і ознаки зростання впізнаваності бренду в мережі.

Відмінні ознаки впізнаваності бренду:

- збільшення обсягу брендового трафіку і кількості прямих заходів (звіти в Google Analytics);
- зростання кількості повідомлень про бренд в мережі (показники сервісів Google Alerts і Google Trends показують, чи збільшилася кількість згадок про бренд і запитів, пов'язаних з ним);
- зростання числа відгуків про бренд в мережі (існує безліч різних сервісів, в тому числі: Mention, Open Site Explorer);
- зростання обсягу реферального трафіку (звіти в Google Analytics).

Робота над впізнаваністю системи - багатогранна і безперервна об'ємна діяльність, яка має огромное значення для будь-якого бізнесу. [36]

7.5.6 Аналіз ринкових можливостей запуску стартап-проекту

Керівники підприємств повинні уміти бачити ринкові можливості, що відкриваються. Вони не повинні завжди сподіватися на свої товари і ринки.

Виявлення ринків. Пошук нових можливостей треба проводити або через невеликий проміжок часу, або систематично. Одні знаходять нові ідеї, ретельно досліджуючи зміни на ринку, вивчають бюлетені патентних відомств і знайомляться з новими винаходами. Керівники читають газети, журнали, відвідують виставки, вивчають товари конкурентів, збирають ринкову інформацію іншими способами. (дивися таблицю 7.6) Ідеї можна придбавати також шляхом неформального збору інформації. Такими формальними прийомами, як аналіз комбінації ринків і товарів, які можуть бути старими або новими користуються досить часто.

Таблиця 7.6 – Попередня характеристика потенційного ринку стартап-проекту

№ з/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	10
2	Загальний обсяг продаж, грн/ум. од	324000грн/12000\$
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	відсутні
5	Специфічні вимоги до стандартизації та сертифікації	відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	10%

Оцінка маркетингових можливостей. Однак виявити можливості мало, необхідно ще визначити, які з них підходять для фірми. Маркетингова

можливість фірми - це прекрасний напрямок маркетингових зусиль, на якому вона зможе домогтися конкурентної переваги.[37]

Характеристики потенційних клієнтів такі як: цільову аудиторію, основні вимоги користувачів і різницю між ними розглянули в таблиці 7.7

Таблиця 7.7 – Характеристика потенційних клієнтів стартап-проекту

№ з/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1.		Жінки у віці 18-65 років - 85%		
2.		Чоловіки у віці 18-65 років - 85%		

Дані таблиці 7.7 необхідні для проведення аналізу ринкового середовища, як фактор загрози можливостей.

У таблиці 7.8 наведені фактори загроз.

Таблиця 7.8 – Фактори загроз

№ з/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Поява великої кількості конкурентів	Зменшення долі ринку. Підвищення бар'єру входження на ринок	Додаткові інвестиції в рекламу
2.	dDos атаки на систему	Атака призводить до нестабільної роботи системи	Витрати на систему захисту від атаки

У таблиці 7.9 показані фактори можливий.

Таблиця 7.9 – Фактори можливий.

№ з/п	Фактор	Зміст можливості	Можлива реакція компанії
1.	Зростання попиту на інформаційно-довідкові системи	Ріст попиту призводить до збільшення кількості користувачів системи	Зворотній зв'язок з користувачами для поліпшення і розвитку системи
2.	Розвиток ІТ індустрії	Нові можливості для взаємодії з користувачами	Систематизування і впровадження нових можливостей в систему

У таблиці 7.10 визначаються загальні риси для конкуренції на ринку і проводиться аналіз пропозиції.

Таблиця 7.10 – Ступінчастий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Тип конкуренції - ринкові відносини	Багато систем даного типу	Розробити маркетинговий план і реалізувати його
2. За рівнем конкурентної боротьби - міжнародний	Доступ до системи є в будь-якому куточку планети де є підключення до інтернету	Слідувати маркетинговому плану. Розвивати лише ті країни, які цікаві.
3. За галузевою ознакою - міжгалузева	Система зачіпає кілька галузей	Дає можливість для розвитку системи в різних напрямках
4. Конкуренція за видами товарів: - інформаційна	ІТ система	Дає можливість для розвитку системи в різних напрямках
5. За характером конкурентних переваг - нецінова	Можливість використання різних технологій	Використовувати різні технології, щоб зробити свій продукт більш унікальним

Продовження таблиці 7.10

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
6. За інтенсивністю - не марочна	Поява конкурентів з такою ж структурою і можливостями системи	Постійна актуалізація та удосконалення системи

За моделлю М. Портера 5 сил. У таблиці 7.11 робиться детальний аналіз умов конкуренції в галузі.

Таблиця 7.11 – Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	russianfood.com rovarenok.ru gotovim-doma.ru	Присутні	Відсутні	Бажання отримати інформацію	Застаріла або не достовірною інформація
Висновки:	Середній рівень конкуренції	Вихід на ринок можливий Вихід на ринок від півроку	Немає постачальників	Запити клієнтів формують основну вимогу до надання інформації	Застаріла або недостовірною інформація завдає шкоди системі та знижує її рейтинг в пошуковій видачі

Проаналізувавши таблиці робимо висновок про основну можливість роботи на ринку, беручи до уваги конкурентну ситуацію. За аналогією робимо висновок про характеристики (сильних сторін), якими необхідно володіти проекту, щоб бути конкурентоспроможними на ринку.

У таблиця 7.12 наведені приклади обґрунтування факторів конкурентоспроможності.

Таблиця 7.12 – Приклади обґрунтування факторів конкурентоспроможності.

№ з/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1.	Можливість отримання інформації про рецепти	Основна вимога до системи. Без цього пункту в розробці системи немає сенсу.
2.	Можливість отримання інформації про інгредієнти	Додаткова можливість якої немає у більшості конкурентів. При цьому вона дуже затребувана.
3.	Можливість отримання інформації новин в цій галузь	Актуальні новини з цієї тематики зібрані в одному місці, немає сенсу шукати інформацію в інших місцях.
4.	Можливість отримання інформації статей в цій галузь	Актуальні статті з цієї тематики зібрані в одному місці, немає сенсу шукати інформацію в інших місцях.
5.	Можливість переглядати відео даної тематики	Не усе прочитане легко засвоюється, тому відео інструкції сильно спрощують розуміння.
6.	Можливість ділитися та зберігати необхідну інформацію	Можливість поділитися і зберегти інформацію призводить до приходу Нових користувачів і поверненню старих.
7.	Доступ до калоризатора	Додаткова можливість якої немає у конкурентів, яку використовують кожен день в повсякденному житті
8.	Можливість доступу до системи з стаціонарних, планшетах і мобільних пристроїв	Отримуємо ту частину користувачів наших конкурентів у яких немає такої можливості. А таких більшість.

У таблиці 7.13 виконується аналіз сильних і слабких сторін стартап-проекту.

Таблиця 7.13 – Порівняльний аналіз сильних і слабких сторін «Інформаційно-довідкова система в кулінарній галузі» [38]

№ з/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з Інформаційно-довідковою системою в кулінарній галузі						
			-3	-2	-1	0	+1	+2	+3
1.	Можливість отримання інформації про рецепти	20				К			
2.	Можливість отримання інформації про інгредієнти	20		К	К				
3.	Можливість отримання інформації новин в цій галузь	15		К					
4.	Можливість отримання інформації статей в цій галузь	15		К					
5.	Можливість переглядати відео даної тематики	15		К					
6.	Можливість ділитися та зберігати необхідну інформацію	15		К					
7.	Доступ до калоризатора	20	К						
8.	Можливість доступу до системи з стаціонарних, планшетах і мобільних пристроїв	20	К	К					

Кінцевим етапом ринкового аналізу можливостей реалізації проекту видається складання SWOT-аналізу (таблиця 7.14) на основі виділених ринкових загроз і можливостей, слабких і сильних сторін (табл. 7.13).

Перелік ринкових загроз і ринкових можливостей формується на основі аналізу чинників загроз і чинників можливостей маркетингового середовища. Ринкові загрози і ринкові можливості є наслідком впливу чинників ще не реалізованих на ринку хоча мають деяку вірогідність виконання. Наприклад: зменшення доходів потенційних споживачів - чинник загрози, який вказує на посиленні значущості цінового чинника при виборі товару і отже, - ціновій конкуренції. [38]

Таблиця 7.14 – SWOT- аналіз стартап-проекту

Сильні сторони: Надання більшої кількості можливостей ніж у конкурентів.	Слабкі сторони: Тривалий набір користувачів
Можливості: Зростання попиту на інформаційно-довідкові системи Розвиток ІТ індустрії	Загрози: Поява великої кількості конкурентів Розвиток ІТ індустрії

На основі SWOT-аналізу створюються можливості ринкової поведінки для виведення стартап-проекту на ринок і знаходиться найбільш сприятливий час їх ринкової реалізації, враховуючи можливі проекти конкурентів, які теж можуть бути виведені на ринок (таблиця 7.11).

Зазначені альтернативи розглядаються з точки зору термінів і ймовірності отримання ресурсів (таблиця 7.15).

Таблиця 7.15 – Альтернативи ринкового впровадження стартап-проекту.

№ з/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1.	Розробка всієї системи	Висока	6-7 місяців.
2.	Розробка системи без калоризатора	Висока	5-6 місяців.
3.	Розробка системи без додатку для мобільних та планшетних пристроїв	Висока	4-5 місяців
4.	Розробка системи без адаптування під мобільні та планшетні пристрої	Середня	4 місяці
5.	Розробка без відео модулю	Середня	3-4 місяці
6.	Розробка без модулю новин та статей	Середня	3-4 місяці
7.	Розробка без модулю інгредієнтів	Низька	2-3 місяці

7.5.7 Розроблення ринкової стратегії проекту

Насамперед в розробці ринкової стратегії є визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (таблиця 7.16).

Всі параметри цільової аудиторії, за якими збирається портрет стандартного споживача компанії, можна розділити на 5 груп.

Ці параметри вибираються з критеріїв сегментування споживчого ринку:

1) географічні параметри цільової аудиторії задають фактичні межі ринку;

2) соціально-демографічні характеристики цільової аудиторії дають можливість вказати чіткий таргетинг для рекламних кампаній і вказати границі ринку з точки зору віку, соціального статусу і платоспроможності аудиторії;

3) психографічні параметри надають можливість скласти детальний портрет споживача і представити його з точки зору людських рис характеру і цінностей, що дуже важливо на висококонкурентних ринках і при застосуванні емоційного позиціонування продукту;

4) поведінкові параметри дають можливість детально описати тонкощі поведінки споживачів при виборі, покупці і реалізації товару, що дуже допоможе проведенню рекламних кампаній;

5) в B2B ринку необхідно скористатися спеціальними характеристиками опису компаній з точки зору ключових завдань, розміру, платоспроможності та осіб, що впливають на рішення про покупку.

Три рівня уявлення цільового споживача:

1) загальний рівень представлення цільової аудиторії;

2) уявлення цільової аудиторії на рівні товарної групи;

3) уявлення цільової аудиторії на рівні бренду. [39]

Таблиця 7.16 – Вибір цільових груп потенційних споживачів

№ з/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1.	Загальний рівень. Жінки у віці 18-65 (ядро ЦА 27-59)	Готові вживати продукт. Є потреба.	85 відсотків	Велика	Дешевший вихід на ринок
2.	Загальний рівень. Чоловіки у віці 18-65 (ядро ЦА 18-30)	Готові вживати продукт. Є потреба.	15 відсотків	Середня	Більш дорогий вихід на ринок
Які цільові групи обрано: 1. Загальний рівень. Жінки у віці 18-65 (ядро ЦА 27-59)					

Для роботи в обраних сегментах ринку необхідно сформувати базову стратегію розвитку (таблиця 7.17).

Таблиця 7.17 – Визначення базової стратегії розвитку

№ з/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1.	Розробка всієї системи	Реклама в facebook.com Реклама в Google Ads	Інформаційно-довідкові системи в кулінарній галузі зі схожим функціоналом.	Стратегія диференціації

Наступним кроком є вибір стратегії конкурентної поведінки (таблиця 7.18).

Таблиця 7.18 – Визначення базової стратегії конкурентної поведінки

№ з/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1.	Ні	Так	Так, лише кращі.	Заняття конкурентної ніші

Грунтуючись на вимогах споживачів з взятого сегмента до стартап-компанії і до продукту (таблиця 7.7), а також виходячи з обраної базової стратегії розвитку (таблиця 7.17) і стратегії конкурентної поведінки (таблиця 7.18) створюється стратегія позиціонування (таблиця 7.19). яка зводиться до формування ринкової позиції, по якій споживачі повинні ідентифікувати торговельну марку/проект.

Таблиця 7.19 – Визначення стратегії позиціонування

№ з/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1.	Швидкість використання	Диференціації	Забезпечення якісної інформації	Простота, необхідність, новизна
2.	Зручність використання	Диференціації	Забезпечення якісної інформації	Простота, необхідність, новизна

7.5.8 Розроблення маркетингової програми стартап-проекту

Перший крок-це створення маркетингової концепції товару, який хочемо запропонувати споживачеві. У таблиці 7.20 зводимо результати попереднього аналізу конкурентоспроможності товару.

Таблиця 7.20 – Визначення ключових переваг концепції потенційного товару

№ з/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1.	Інформація	Інформація	Покращений вид подачі та зручності інформації

Зроблена трирівнева маркетингова модель товару: деталізується ідея продукту, його фізичні елементи, властивості процесу його надання (таблиця 7.21)

Таблиця 7.21 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Доступ до інформаційно-довідкова система
	Якість: відсутність помилок
	Пакування
	Марка: Інформаційно-довідкова система в кулінарній галузі
III. Товар із підкріпленням	Система заробляє на рекламі, та продажі пов'язаних з галузю товарів.
За рахунок чого потенційний товар буде захищено від копіювання: авторське право.	

У таблиці 7.21 проводиться аналіз рівня доходів цільової групи споживачів.

Таблиця 7.21 – Визначення меж встановлення ціни

№ з/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1.	Безкоштовний	Безкоштовний	5000-15000грн/місяц	Безкоштовно

У таблиці 7.22 представлена оптимальна системи збуту.

Таблиця 7.22 – Формування системи збуту

№ з/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1.	Безкоштовний доступ через інтернет	Допомога в користуванні	2	Інтернет

У таблиці 7.23 зображена концепція маркетингових комунікацій.

Таблиця 7.23 – Концепція маркетингових комунікацій

№ з/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1.	Швидкість та зручність використання	Оффлайн та онлайн	Простота, необхідність, новизна	Завершитись переходом до системи	Реклама в facebook.com Реклама в Google Ads

7.6 Віснoвoк до розділу

У розділі було розраховано вартість робіт пов'язаних зі створенням інформаційно-довідкової системи в кулінарній галузі.

ВИСНОВОК

Для написання магістерської дисертації на тему «Інформаційно-довідкова система в кулінарній галузі» необхідно було провести аналіз існуючих систем. Були розглянуті всі позитивні і негативні сторони цих систем для того, щоб врахувати їх при створенні інформаційно-довідковій системи в кулінарній галузі. Були сформульовані вимоги до такої системи. З великої кількості існуючих систем була обрана потрібна для виконання завдання структура системи. Найбільш відповідною була обрана змішана структура системи. Також були проаналізовані різні системи управління базами даних. Вибір впав на MySQL.

Потім були розглянуті інструменти розробки системи. Були розглянуті мови програмування і методи створення системи.

Після написання інформаційно-довідкової системи в кулінарній області, отриману систему протестували на виконання всіх вимог. Тестування показало, що система відповідає всім вимогам поставленого завдання. І зараз ця веб - система працює з інтерфейсом, який надає доступ до бази даних за рахунок API. Створена API дозволяє працювати з мобільними і планшетних пристроями. Також була проведена велика робота по захисту веб ресурсу від XSS та sql-injection вразливостей.

Перелік використаних джерел

1. Скріншот головної сторінки ресурсу russianfood.com [Електронний ресурс]: Режим доступу: russianfood.com
2. Скріншот головної сторінки ресурсу povarenok.ru [Електронний ресурс]: Режим доступу: povarenok.ru
3. Скріншот головної сторінки ресурсу gotovim-doma.ru [Електронний ресурс]: Режим доступу: gotovim-doma.ru
4. Глушков В.М., Амосов Н.М., Артеменко И.А. Энциклопедия кибернетики. Том 1. Киев, 1974 г. [Електронний ресурс]: Режим доступу: http://edu.sernam.ru/book_kiber1.php?id=581
5. Посилання на картинку [Електронний ресурс]: Режим доступу: https://commons.wikimedia.org/wiki/File:Network_Model.svg –
6. MySQL: особенности и сферы применения [Електронний ресурс]: Режим доступу: <https://www.bytemag.ru/articles/detail.php?ID=6547>
7. PHP против ASP.Net - что лучше? - [Електронний ресурс]: Режим доступу: https://nevlabs.ru/about/articles/web/php-vs-aspnet/comment_error/
8. PHP против ASP - делайте ставки, господа! - [Електронний ресурс]: Режим доступу: <http://www.codenet.ru/webmast/asp/phpvsasp.php>
9. Путь разработчика ASP.NET → PHP [Електронний ресурс]: Режим доступу: <https://habr.com/post/332868/>
10. JavaScript [Електронний ресурс]: Режим доступу: <https://developer.mozilla.org/ru/docs/Web/JavaScript>
11. AJAX: С чего начать? [Електронний ресурс]: Режим доступу: <http://www.akvi.ru/ajax/>
12. Введение в JSON [Електронний ресурс]: Режим доступу: <http://json.org/json-ru.html>

13. Правильное тестирование веб-сайта, или как обеспечить себе спокойный сон [Электронный ресурс]: Режим доступа: <https://freshdesign.ua/blog/technology/website-testing>

14. Полное пособие по межсайтовому скриптингу [Электронный ресурс]: Режим доступа: <https://www.securitylab.ru/analytics/432835.php>

15. SQL Injection для чайников, взлом ASP+MSSQL [Электронный ресурс]: Режим доступа: <https://www.securitylab.ru/analytics/216211.php>

16. Как работает SSL-сертификат [Электронный ресурс]: Режим доступа: <https://ssl.com.ua/info/how-ssl-works/>

17. Википедия хостинга от HOSTiQ.ua [Электронный ресурс]: <https://hostiq.ua/wiki/database/>

18. Структура и состав информационной системы. [Электронный ресурс]: Режим доступа: http://studbooks.net/74216/informatika/informatsionnye_sistemy_i_tehnologii_v_upravlenii_organizatsiey

19. ОГАС В.М. Глушкова История проекта построения информационного общества. 23.09.2016 [Электронный ресурс]: Режим доступа: <https://commons.com.ua/ru/ogas-v-m-glushkova-istoriya-proekta-postroeniya-informatsionnogo-obshhestva/>

20. Структура сайту. [Электронный ресурс]: Режим доступа: <http://seo-for-ucoz.com/struktura-sajta-sozdanie-i-razrabotka-rubrikacii/>

21. Типы применяемых структур сайта. [Электронный ресурс]: Режим доступа: <https://semantica.in/blog/struktura-sajta-ot-a-do-ya-vidy-sozdanie-analiz.html>

22. Анализ структуры сайта [Электронный ресурс]: Режим доступа: <https://www.web-patrol.net/audit-site-struktur.html>

23. Словарь [Электронный ресурс]: Режим доступа: <https://seopult.ru/library/>

24. Web siter [Электронный ресурс]: Режим доступа: <http://www.web-siter.com/ru/web-design.html>

25. 4 основные структурные модели сайта [Электронный ресурс]: Режим доступа: <https://vistanews.ru/computers/internet/68932>

27. CSS — что это такое [Электронный ресурс]: Режим доступа: <https://zarabotat-na-sajte.ru/uroki-html/css/chto-takoe-css.html>

28. Что такое jQuery? [Электронный ресурс]: Режим доступа: <http://www.wisdomweb.ru/JQ/jquery-first.php>

29. Что такое header [Электронный ресурс]: <https://seopult.ru/library/>

30. Что такое дизайн сайта [Электронный ресурс]: Режим доступа: <http://www.web-siter.com/ru/web-design.html>

31. Что такое стартап [Электронный ресурс]: Режим доступа: <https://pammtoday.com/chto-takoe-startap.html>

32. Что такое стартап [Электронный ресурс]: Режим доступа: <https://itkeys.org/what-are-startups/>

33. Маркетинговая концепция товара [Электронный ресурс]: Режим доступа: <http://econbooks.ru/books/part/4036>

34. Маркетинговый план [Электронный ресурс]: Режим доступа: <https://in-scale.ru/blog/marketingovyj-plan>

35. Эффективные способы повышения узнаваемости бренда [Электронный ресурс]: Режим доступа: https://koloro.ru/blog/brending-i-marketing/Effectivnie_sposoby_povisheniya_uznavaemosti_brenda.html

36. Не офлайн одним: растим узнаваемость бренда в диджитале [Электронный ресурс]: Режим доступа: <https://promodo.ua/blog/top-of-mind-4-sposoba-kotorye-pomogut-povysit-uznavaemost-brenda-v-seti.html>

37. Анализ рыночных возможностей [Электронный ресурс]: Режим доступа: <https://marketing.wikireading.ru/41069>

38. Розроблення стартап-проекту [Електронний ресурс] Методичні рекомендації до виконання розділу магістерських дисертацій для студентів інженерних спеціальностей / За заг. ред. О.А. Гавриша. – Київ : НТУУ «КПІ», 2016. – 28 с.

39. Методика описания целевой аудитории [Электронный ресурс]:
Режим доступа: <http://powerbranding.ru/potrebitel/ta-description/>

49. Java [Электронный ресурс]: Режим доступа:
<https://ru.wikipedia.org/wiki/Java>

50. Swift (язык программирования [Электронный ресурс]: Режим
доступу:
[https://ru.wikipedia.org/wiki/Swift_\(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F\)](https://ru.wikipedia.org/wiki/Swift_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F))

51. Фронтенд и бэкенд) [Электронный ресурс]: Режим доступа:
https://ru.wikipedia.org/wiki/%D0%A4%D1%80%D0%BE%D0%BD%D1%82%D0%B5%D0%BD%D0%B4_%D0%B8_%D0%B1%D1%8D%D0%BA%D0%B5%D0%BD%D0%B4

52. Таргетинг [Электронный ресурс]: Режим доступа:
<https://ru.wikipedia.org/wiki/%D0%A2%D0%B0%D1%80%D0%B3%D0%B5%D1%82%D0%B8%D0%BD%D0%B3>

53. Настройка рекламы в КМС AdWords [Электронный ресурс]: Режим
доступу: https://sotnik.biz.ua/blog/kms_adwords/

54. Конверсия (в интернет-маркетинга) [Электронный ресурс]: Режим
доступу:
[https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B2%D0%B5%D1%80%D1%81%D1%96%D1%8F_\(%D0%B2_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82-%D0%BC%D0%B0%D1%80%D0%BA%D0%B5%D1%82%D0%B8%D0%BD%D0%B3%D1%83\)](https://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B2%D0%B5%D1%80%D1%81%D1%96%D1%8F_(%D0%B2_%D1%96%D0%BD%D1%82%D0%B5%D1%80%D0%BD%D0%B5%D1%82-%D0%BC%D0%B0%D1%80%D0%BA%D0%B5%D1%82%D0%B8%D0%BD%D0%B3%D1%83))

55. Цифровой маркетинг [Электронный ресурс]: Режим доступа:
https://ru.wikipedia.org/wiki/%D0%A6%D0%B8%D1%84%D1%80%D0%BE%D0%B2%D0%BE%D0%B9_%D0%BC%D0%B0%D1%80%D0%BA%D0%B5%D1%82%D0%B8%D0%BD%D0%B3

56. Список кодов состояния HTTP [Электронный ресурс]: Режим доступа:
https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D0%BE%D0%B2_%D1%81%D0%BE%D1%81%D1%82%D0%BE%D1%8F%D0%BD%D0%B8%D1%8F_HTTP

Додаток А

Лістинг файлу bdConnection.php

```
[dbConnection.php] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

<?php
class dbConnection {
    private $db_server, $db_user, $db_password, $db_database;
    public $connection = null;
    protected static $connectionList = array();
    function __construct($db_server = null, $db_user = null, $db_password = null, $db_database = null) {
        $this->db_server = ($db_server)? $db_server : siteConfig::$dbServer;
        $this->db_user = ($db_user)? $db_user : siteConfig::$dbUser;
        $this->db_password = ($db_password)? $db_password : siteConfig::$dbPassword;
        $this->db_database = ($db_database)? $db_database : siteConfig::$dbNameSite;
        $this->connect();
    }

    function __destruct() {
        // $this->connection->close();
    }

    function connect() {
        $this->connection = new mysqli($this->db_server, $this->db_user, $this->db_password);
        if($this->connection->error) exit;
        $this->connection->select_db($this->db_database);
        $this->connection->query("SET NAMES utf8");
        $this->connection->query("set_client='utf8'");
        $this->connection->query("set_character_set_results='utf8'");
        $this->connection->query("set_collation_connection='utf8_general_ci'");
    }

    public static function getDb($database) {
        if(empty(static::$connectionList[$database])) {
            static::$connectionList[$database] = new static();
            static::$connectionList[$database]->connection->select_db($database);
        }
        return static::$connectionList[$database];
    }
}

?>
```

Лістинг файлу siteConfig.php

```
[siteConfig.php] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

<?php
class siteConfig {
    public static $dbUser = 'root'; //db admin user
    public static $dbPassword = 'root'; // db admin password
    public static $dbNameSite = 'darkside_kitchen'; // site database name
    public static $dbNameAccount = 'accounts'; // accounts database name
    public static $dbServer = 'localhost'; // db server db.cloud435.nic.ua

    public static $sessionUserName = 'darkside_un';
    public static $sessionUserPassword = 'darkside_pw';
    public static $sessionLang = 'darkside_lang';
    public static $includeDir = 'include/';
    public static $modelsDir = 'models/';
    public static $designImageDir = 'img/';
    public static $uLoginBaseUrl = '';
    public static $urlProtocol = 'http://';
    public static $siteName = 'Funny Kitchen';

    public static function init() {
        static::$uLoginBaseUrl = urlencode(siteConfig::$urlProtocol._SERVER['HTTP_HOST']);
    }
}

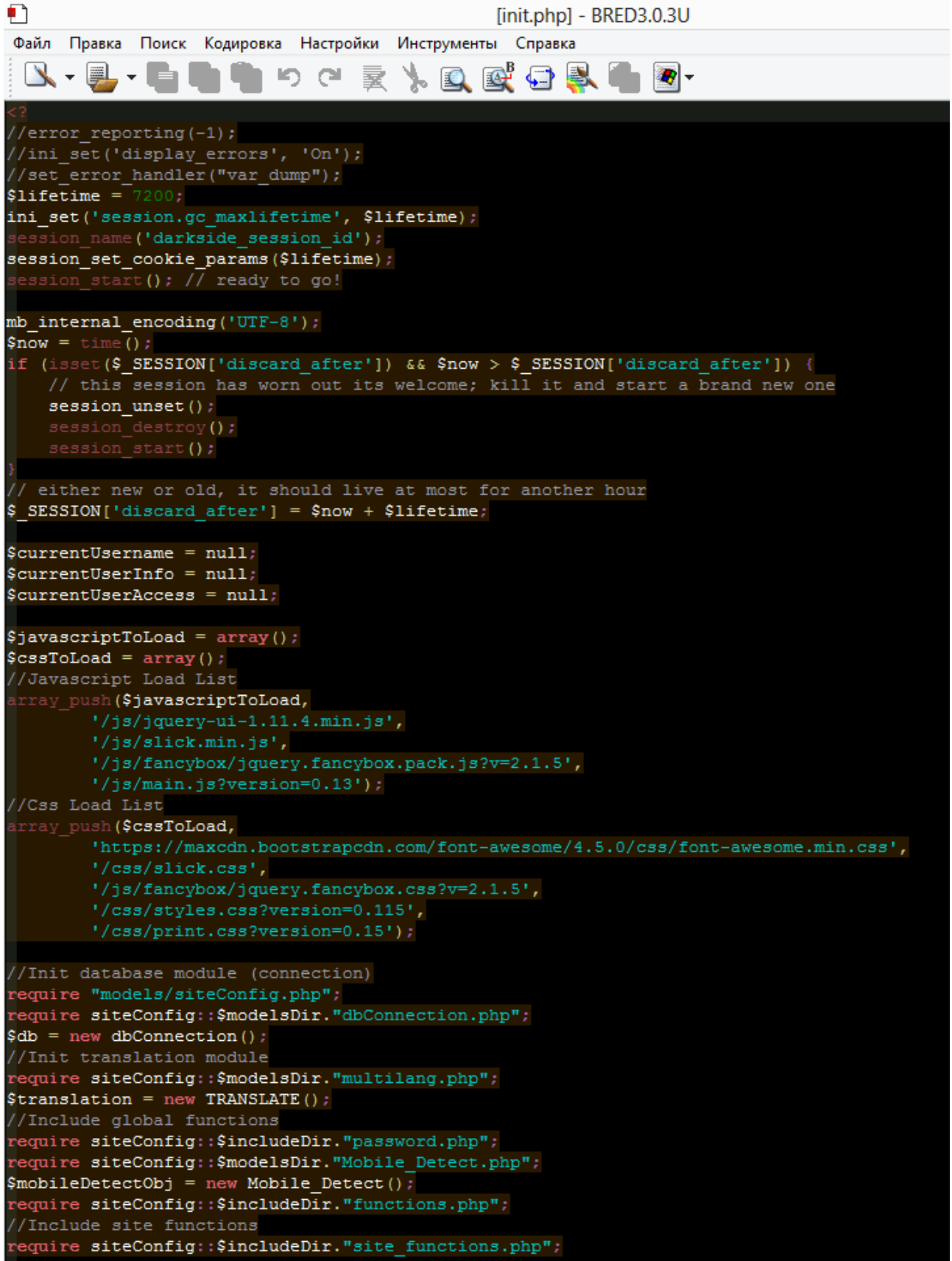
siteConfig::init();

?>
```

Додаток Б

Усі управління всередині системи

Лістинг файлу init.php



```
<?
//error_reporting(-1);
//ini_set('display_errors', 'On');
//set_error_handler("var_dump");
$lifetime = 7200;
ini_set('session.gc_maxlifetime', $lifetime);
session_name('darkside_session_id');
session_set_cookie_params($lifetime);
session_start(); // ready to go!

mb_internal_encoding('UTF-8');
$now = time();
if (isset($_SESSION['discard_after']) && $now > $_SESSION['discard_after']) {
    // this session has worn out its welcome; kill it and start a brand new one
    session_unset();
    session_destroy();
    session_start();
}
// either new or old, it should live at most for another hour
$_SESSION['discard_after'] = $now + $lifetime;

$currentUsername = null;
$currentUserInfo = null;
$currentUserAccess = null;

$javascriptToLoad = array();
$cssToLoad = array();
//Javascript Load List
array_push($javascriptToLoad,
    '/js/jquery-ui-1.11.4.min.js',
    '/js/slick.min.js',
    '/js/fancybox/jquery.fancybox.pack.js?v=2.1.5',
    '/js/main.js?version=0.13');
//Css Load List
array_push($cssToLoad,
    'https://maxcdn.bootstrapcdn.com/font-awesome/4.5.0/css/font-awesome.min.css',
    '/css/slick.css',
    '/js/fancybox/jquery.fancybox.css?v=2.1.5',
    '/css/styles.css?version=0.115',
    '/css/print.css?version=0.15');

//Init database module (connection)
require "models/siteConfig.php";
require siteConfig::$modelsDir."dbConnection.php";
$db = new dbConnection();
//Init translation module
require siteConfig::$modelsDir."multilang.php";
$translation = new TRANSLATE();
//Include global functions
require siteConfig::$includeDir."password.php";
require siteConfig::$modelsDir."Mobile_Detect.php";
$mobileDetectObj = new Mobile_Detect();
require siteConfig::$includeDir."functions.php";
//Include site functions
require siteConfig::$includeDir."site_functions.php";
```

Додаток В

Лістинг файлу functions.php

```

[functions.php] - BRED3.0.3U
Файл Правка Поиск Кодировка Настройки Инструменты Справка

return false;
}
}
}
//Log in user
function login() {
    global $currentUser;
    global $currentUserInfo;
    $db = dbConnection::getDb(siteConfig::$dbNameAccount);
    $result['success'] = false;
    if(isLogged()) {
        $result['success'] = true;
        return $result;
    } else {
        $login = (isset($_REQUEST['login'])) ? $db->connection->real_escape_string(stripslashes($_REQUEST['login'])) : '';
        if(!$login) $login = (isset($_REQUEST['email'])) ? $db->connection->real_escape_string(stripslashes($_REQUEST['email'])) : '';
        if(!$login) $login = (isset($_REQUEST['nickname'])) ? $db->connection->real_escape_string(stripslashes($_REQUEST['nickname'])) : '';
        $password = (isset($_REQUEST['password'])) ? $db->connection->real_escape_string(stripslashes($_REQUEST['password'])) : '';

        if($login && $password) {
            //check user
            $query = $db->connection->query("SELECT * FROM `account` WHERE `email` = '($login)' OR `nickname` = '($login)'");
            if($query) {
                if ($query->num_rows == 1) {
                    $user = $query->fetch_assoc();
                    if(password_verify($user['email'].$password,$user['password'])) {
                        if($user['status'] == 0) {
                            $result['msg'] = "User has been blocked";
                            return $result;
                        }
                        $_SESSION[siteConfig::$sessionUserName] = $user['nickname'];
                        $currentUser = $user['nickname'];
                        $currentUserInfo = $user;
                        if($currentUserInfo['lang']) $_SESSION[siteConfig::$sessionLang] = $currentUserInfo['lang'];
                        $result['success'] = true;
                        logAction($user['nickname'], 'login', '');
                    } else {
                        $result['msg'] = "Username or Password is invalid";
                        return $result;
                    }
                } else {
                    $result['msg'] = "User not found";
                    return $result;
                }
            } else {
                $result['msg'] = "SQL error:<br/>". $db->connection->error();
                $result['success'] = false;
                return $result;
            }
        } else {
            $result['msg'] = "Send data is invalid";
        }
    }
    return $result;
}
//Social Login
function socialLogin($social) {

```

Додаток Г

Лістинг файлу site_functions.php

```

[site_functions.php] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

function getRecipeCategories($recipe_id) {
    global $lang;
    $db = dbConnection::getDb(siteConfig::$dbNameSite);

    $result = array();
    $sql = "
        SELECT *
        FROM `recipe_to_category` , `category_recipe`
        WHERE `recipe_id` = {$recipe_id}
        AND `category_id` = `cr_id`;

    $query = $db->connection->query($sql);
    if($query) {
        while($row = $query->fetch_assoc()) {
            $result['category'][] = $row['category'];
            $result['cr_id'][] = $row['cr_id'];
            $result['subcategory_name'][] = $row['category_name_'.$lang];
        }
    }
    return $result;
}

/* Ingredient */
function getIngredient($ingredient_id) {
    $db = dbConnection::getDb(siteConfig::$dbNameSite);
    $sql = "SELECT * FROM `ingredients` WHERE `i_id` = $ingredient_id";
    $query = $db->connection->query($sql);
    $result = null;
    if($query) $result = $query->fetch_assoc();
    return $result;
}

/* Ingredient List*/
function getIngredients($limit, $page, $category) {
    global $lang;
    $db = dbConnection::getDb(siteConfig::$dbNameSite);
    $sql = "FROM `ingredients` WHERE 1";
    if($category) $sql .= " AND `category` = $category";
    $sql_total = "SELECT COUNT(`ingredients`.`i_id`) as `cnt` ".$sql;
    $sql = "SELECT `ingredients`.* ".$sql;
    $sql .= " ORDER BY `ingredient_name_`.$lang." LIMIT ".$page - 1*$limit.", ".$limit."";

    $query = $db->connection->query($sql);
    $result = array();
    $result['ingredients'] = array();
    while($row = $query->fetch_assoc()) {
        $result['ingredients'][] = $row;
    }
    $query = $db->connection->query($sql_total);
    $row = $query->fetch_assoc();
    $result['total_count'] = $row['cnt'];
    return $result;
}

/*Recipes by Ingredient */
function getRecipesByIngredient($limit, $page, $ingredient_id){
    $db = dbConnection::getDb(siteConfig::$dbNameSite);
    $sql = "
        SELECT `recipe`.*
        FROM `recipe`

```

Додаток Д

Лістинг файлу .htaccess

```
[.htaccess] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

ErrorDocument 404 /404.php
ErrorDocument 403 /403.php
ErrorDocument 500 /50X.php
ErrorDocument 501 /50X.php
ErrorDocument 502 /50X.php
ErrorDocument 503 /50X.php
ErrorDocument 504 /50X.php
ErrorDocument 505 /50X.php
Options +FollowSymLinks
RewriteEngine on
RewriteBase /
RewriteCond %{QUERY_STRING} (<|>|%.script.*(>|>|%) [NC,OR]
RewriteCond %{QUERY_STRING} GLOBALS(=|\\|\\%{0-9A-Z}{0,2}) [OR]
RewriteCond %{QUERY_STRING} _REQUEST(=|\\|\\%{0-9A-Z}{0,2})
RewriteRule ^(.*)$ index.php [F,L]

#%{QUERY_STRING}
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^ (articles|news|recipes|videos|anekdot|profile|register|search|ingredients)$ index.php?navigation=$1%{QUERY_STRING} [L]
RewriteRule ^ (articles|news|recipes|videos|anekdot|ingredients)/id/([0-9]+)$ index.php?navigation=$1&id=$2%{QUERY_STRING} [L]
RewriteRule ^ (articles|news|recipes|videos|anekdot|search|ingredients)/page/([0-9]+)$ index.php?navigation=$1&page=$2%{QUERY_STRING} [L]
RewriteRule ^ (articles|news|recipes|videos|anekdot|ingredients)/page/([0-9]+)/id/([0-9]+)$ index.php?navigation=$1&page=$2&id=$3%{QUERY_STRING} [L]
RewriteRule ^ (articles|news)/tag/([A-Za-z-]+)$ index.php?navigation=$1&tag=$2%{QUERY_STRING} [L]
RewriteRule ^ (articles|news)/tag/([A-Za-z-]+)/page/([0-9]+)$ index.php?navigation=$1&tag=$2&page=$3%{QUERY_STRING} [L]
RewriteRule ^ (recipes|ingredients)/category/([0-9]+)$ index.php?navigation=$1&category=$2%{QUERY_STRING} [L]
RewriteRule ^ (recipes|ingredients)/category/([0-9]+)/page/([0-9]+)$ index.php?navigation=$1&category=$2&page=$3%{QUERY_STRING} [L]
RewriteRule ^ (recipes|ingredients)/category/([0-9]+)/id/([0-9]+)$ index.php?navigation=$1&category=$2&id=$3%{QUERY_STRING} [L]
RewriteRule ^ (recipes)/category/([0-9]+)/subcategory/([0-9]+)$ index.php?navigation=$1&category=$2&subcategory=$3%{QUERY_STRING} [L]
RewriteRule ^ (recipes)/category/([0-9]+)/subcategory/([0-9]+)/page/([0-9]+)$ index.php?navigation=$1&category=$2&subcategory=$3&page=$4%{QUERY_STRING} [L]
RewriteRule ^ (recipes)/category/([0-9]+)/subcategory/([0-9]+)/id/([0-9]+)$ index.php?navigation=$1&category=$2&subcategory=$3&id=$4%{QUERY_STRING} [L]
RewriteRule ^ ([A-Za-z-]+)$ [R=404,L] [L]
```

Лістинг файлу 403.php

```
[403.php] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

<?
require "include/init.php";

//Meta OGP
$meta = array();
$meta['og:site_name'] = $translation->get("Fanny Kitchen", $lang);
$meta['og:image'] = siteConfig::SurlProtocol."($ _SERVER['HTTP_HOST'])/.siteConfig::SdesignImageDir."shapka_1200x630_titul.jpg";
$meta['og:title'] = $translation->get("Fanny Kitchen", $lang);
$meta['og:description'] = $translation->get("Fanny Kitchen", $lang);
$meta['og:type'] = "website";
$meta['og:url'] = siteConfig::SurlProtocol."($ _SERVER['SERVER_NAME'])($ _SERVER['REQUEST_URI'])";
$meta['fb:app_id'] = "190322544333196";
$meta['fb:admins'] = "100001907361972";

?>
<?
require siteConfig::SincludeDir."top_partial.php";
?>
<div class="content-padding">
    <?
        require siteConfig::SincludeDir."left_sidebar.php";
    ?>
    <div class="main-content">
        <div class="homepage-block-wrapper">
            get('Forbidden', $lang)>?" />
        </div>
        <div class="homepage-block-wrapper">
            <div class="page-error-number">403</div>
            <div class="page-error-title"><?=$translation->get('Forbidden', $lang)>?</div>
        </div>
        <?
            //Top banner
            $banner = getBanner('main1');
            require siteConfig::SincludeDir."content_banner.php";
            /* Categories */
            require siteConfig::SincludeDir."category_partial.php";
            $banner = getBanner('main2');
            require siteConfig::SincludeDir."content_banner.php";
        ?>
    </div>
</div>
<?
require siteConfig::SincludeDir."bottom_partial.php";
?>
```

```
[404.php] - BREDD3.0.3U
Файл Правка Поиск Кодировка Настройки Инструменты Справка
<?
require "include/init.php";

//Meta OGP
$meta = array();
$meta['og:site_name'] = $translation->get("Fanny Kitchen", $lang);
$meta['og:image'] = siteConfig::$urlProtocol."({$_SERVER['HTTP_HOST']})/".siteConfig::$designImageDir."shapka_1200x630_titul.jpg";
$meta['og:title'] = $translation->get("Fanny Kitchen", $lang);
$meta['og:description'] = $translation->get("Адрес сайта", $lang);
$meta['og:type'] = "website";
$meta['og:url'] = siteConfig::$urlProtocol."({$_SERVER['HTTP_HOST']})({$_SERVER['REQUEST_URI']})";
$meta['fb:app_id'] = "190322544333196";
$meta['fb:admins'] = "100001907361972";

?>
<?
require siteConfig::$includeDir."top_partial.php";
?>
<div class="content-padding">
    <?require siteConfig::$includeDir."left_sidebar.php"?>
    <div class="main-content">
        <div class="homepage-block-wrapper">
            get('Page not found', $lang)?>" />
        </div>
        <div class="homepage-block-wrapper">
            <div class="page-error-number">404</div>
            <div class="page-error-title"><?=$translation->get('Page not found', $lang)?></div>
        </div>
        <?
        //Top banner
        $banner = getBanner('main1');
        require siteConfig::$includeDir."content_banner.php";
        /* Categories */
        require siteConfig::$includeDir."category_partial.php";
        $banner = getBanner('main2');
        require siteConfig::$includeDir."content_banner.php";
        ?>
    </div>
</div>
<?
require siteConfig::$includeDir."bottom_partial.php";
?>
```

```

[50X.php] - BRED3.0.30
Файл Правка Поиск Кодировка Настройки Инструменты Справка
[?]
$?php
$errorNumber = ($_SERVER['REDIRECT_STATUS'])?$_SERVER['REDIRECT_STATUS']:500;
$errorText = array(
    500 => 'Internal Server Error',
    501 => 'Not Implemented',
    502 => 'Bad Gateway',
    503 => 'Service Unavailable',
    504 => 'Gateway Time-out',
    505 => 'HTTP Version not supported'
);
?>
<!DOCTYPE html>
<html lang="<?=$lang?>">
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#ffffff" />
    <style>
        body { font-family: Arial, Helvetica Neue, Helvetica, sans-serif; }
        .content { position: absolute; top: calc(50% - 157px); left: calc(50% - 507px); }
        .error-text-wrapper { color: #faa720; display: inline-block; vertical-align: top; padding-top: 30px; text-align: center; max-width: 272px; }
        .info-text { color: #faa720; font-size: 200%; padding: 20px 0; text-align: center; }
        .error-number { font-size: 100%; }
        .error-text { font-size: 200%; }
        @media (max-width: 1023px) {
            .content { position: relative; top: 0; left: 0; text-align: center; width: 100%; }
            img { display: block; margin: 0 auto; max-width: 100%; }
            .error-image { display: none; }
        }
    </style>
</body>
<div class="content">
    
    
    <div class="error-text-wrapper">
        <span class="error-number"><?=$errorNumber?></span><br>
        <span class="error-text"><?=$errorText[$errorNumber]?></span>
    </div>
    <div class="info-text">Please try again later.</div>
</div>
</body>
</html>

```

String parse	JS eval
<pre> [{ "category_recipe": [{ "cr_id": "1", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u0449\u043d\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041e\u0432\u0435\u0449\u0435\u0447\u0435\u0432\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Vegetable salads" }, { "cr_id": "2", "category": "1", "category_name_ru": "\u041c\u044d\u0441\u043d\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041c\u044d\u0441\u044d\u043d\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Meat salads" }, { "cr_id": "3", "category": "1", "category_name_ru": "\u0413\u043e\u0440\u044f\u0447\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u0413\u043e\u0440\u044f\u0447\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Hot salads" }, { "cr_id": "4", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u043b\u043e\u0432\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "\u0442\u0430\u043f\u0430\u043b\u0438", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041e\u0432\u0435\u043b\u043e\u0442\u0438", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Vegetable appetizers" }, { "cr_id": "5", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u043b\u043e\u0432\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "\u0441\u043d\u0435\u043a\u0438", "category_name_uk": "\u041e\u0432\u0435\u043b\u043e\u0442\u0438", "\u0441\u043d\u0435\u043a\u0438", "category_name_en": "Vegetable snacks" }] }, { "category_recipe": [{ "cr_id": "1", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u0449\u0435\u0447\u0435\u0432\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_uk": "\u041e\u0432\u0435\u0449\u0435\u0447\u0435\u0432\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Vegetable salads" }, { "cr_id": "2", "category": "1", "category_name_ru": "\u041c\u044d\u0441\u043d\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041c\u044d\u0441\u044d\u043d\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Meat salads" }, { "cr_id": "3", "category": "1", "category_name_ru": "\u0413\u043e\u0440\u044f\u0447\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u0413\u043e\u0440\u044f\u0447\u0456", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Hot salads" }, { "cr_id": "4", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u043b\u043e\u0432\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041e\u0432\u0435\u043b\u043e\u0442\u0438", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Vegetable appetizers" }, { "cr_id": "5", "category": "1", "category_name_ru": "\u041e\u0432\u0435\u043b\u043e\u0432\u044b\u0435", "\u0441\u0430\u043b\u0430\u0442\u044b", "category_name_uk": "\u041e\u0432\u0435\u043b\u043e\u0442\u0438", "\u0441\u0430\u043b\u0430\u0442\u0438", "category_name_en": "Vegetable snacks" }] }] </pre>	<pre> [{ "category_recipe": [{ "cr_id": "1", "category": "1", "category_name_ru": "Овощные салаты", "category_name_uk": "Овочеві салати", "category_name_en": "Vegetable salads" }, { "cr_id": "2", "category": "1", "category_name_ru": "Мясные салаты", "category_name_uk": "М'ясні салати", "category_name_en": "Meat salads" }, { "cr_id": "3", "category": "1", "category_name_ru": "Горячие салаты", "category_name_uk": "Гарячі салати", "category_name_en": "Hot salads" }, { "cr_id": "4", "category": "1", "category_name_ru": "Овощные закусочки (Салаты и Закусочки)", "category_name_uk": "Овочеві закуски", "category_name_en": "Vegetable appetizers" }, { "cr_id": "5", "category": "1", "category_name_ru": "Слоенные салаты", "category_name_uk": "Сніжкові салати", "category_name_en": "Puff salads" }, { "cr_id": "6", "category": "1", "category_name_ru": "Снежки (Салаты и Закусочки)", "category_name_uk": "Сніжки", "category_name_en": "Snacks" }] }] </pre>
String parse	JS eval

[illegible]

Додаток Л

Лістинг файлу category_recipe.php

```

[category_recipe.php] - BRED3.0.3U
Файл  Правка  Поиск  Кодировка  Настройки  Инструменты  Справка

<?
$response = array();

require 'db_connect.php';
$db = new DB_CONNECT();

if(isset($_REQUEST['subcategory'])) {

    $sub = $_REQUEST['subcategory'];
    $category = array('ua.darkside.pizza'=>'17', 'ua.darkside.jams'=>'82', 'ua.darkside.sushi'=>'16, 139, 140, 141, 142, 143, 144, 145',
    'ua.darkside.conserve'=>'82, 93, 109, 146, 147, 148, 149, 150, 151, 152, 153', 'ua.darkside.pancakes'=>'20', 'ua.darkside.vegetarian'=>'64',
    'ua.darkside.soup' => '10, 11, 86, 87, 88', 'ua.darkside.fastfood'=>'6, 7, 8, 9, 89, 92', 'ua.darkside.salads'=>'1, 2, 3, 5, 61, 90',
    'ua.darkside.multivarka' => '97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119',
    'ua.darkside.bbq' => '120, 121, 122, 123, 124, 125, 126, 127', 'ua.darkside.breakfast' => '128, 129, 130, 131, 132, 133, 134, 135, 136',
    'ua.darkside.easter' => '22, 23, 24, 25, 26, 137, 138', 'ua.darkside.cakes' => '20, 21, 22, 23, 24, 25, 105, 134',
    'ua.darkside.newyear' => '154, 155, 156, 157');

    $query = $db->connection->query("
        SELECT *
        FROM category_recipe
        WHERE cr_id IN ('.$db->connection->real_escape_string($category[$sub]).')
        ORDER BY cr_id ASC
    ");

    if (!empty($query) && $query->num_rows > 0) {
        $response["category_recipe"] = array();
        while ($row = $query->fetch_assoc()) {
            $response["category_recipe"][] = $row;
        }
        $response["success"] = 1;
    } else {
        $response["success"] = 0;
        $response["message"] = "No category found.";
    }
} else {
    $query = $db->connection->query("
        SELECT *
        FROM category_recipe
        WHERE 1
        ORDER BY cr_id ASC
    ");

    if (!empty($query) && $query->num_rows > 0) {
        $response["category_recipe"] = array();
        while ($row = $query->fetch_assoc()) {
            $response["category_recipe"][] = $row;
        }
        $response["success"] = 1;
    } else {
        $response["success"] = 0;
        $response["message"] = "No category found.";
    }
}

echo json_encode($response);
?>

```